IC 9168

Bureau of Mines Information Circular/1988

MULSIM/BM—A Structural Analysis Computer Program for Mine Design

By Les A. Beckett and Rudy S. Madrid

UNITED STATES DEPARTMENT OF THE INTERIOR

Information Circular 9168

MULSIM/BM—A Structural Analysis Computer Program for Mine Design

By Les A. Beckett and Rudy S. Madrid

UNITED STATES DEPARTMENT OF THE INTERIOR
Donald Paul Hodel, Secretary

BUREAU OF MINES
David S. Brown, Acting Director

Library of Congress Cataloging in Publication Data:

Beckett, Les A.
MULSIM/BM: a structural analysis computer program for mine design.

(Information circular / United States Department of the Interior, Bureau of Mines ; 9168)

Bibliography: p. 16-17.

Supt. of Docs. no.: I 28.27: 9168.

1. Mining engineering—Computer programs. 2. MULSIM/BM (Computer program) I. Madrid, Rudy S. II. Title. III. Series: Information circular (United States. Bureau of Mines) ; 9168.

TN295.U4 [TN292] 622 s [622'.028'55369] 87-600293

CONTENTS

ILLUSTRATIONS

UNIT OF MEASURE ABBREVIATIONS USED IN THIS REPORT

With Factors for Conversion to Units of the International System of Units (SI)

ft	foot	in^2	square inch
GPa	gigapascal	lb	pound
h	hour	lb/in^3	pound per cubic inch
in	inch	pct	percent
in/in	inch per inch	psi	pound per square inch

Abbreviation	Unit of measure	To convert to--	Multiply by--
ft	foot	meters	0.3048 (exactly)
in	inch	centimeters	2.54 (exactly)
psi	pounds per square inch	kilopascals	6.894757

MULSIM/BM—A STRUCTURAL ANALYSIS COMPUTER PROGRAM FOR MINE DESIGN

By Les A. Beckett[1] and Rudy S. Madrid[2]

ABSTRACT

The Bureau of Mines has developed MULSIM/BM, an improved computer program using the boundary-element approach for geomechanical analysis of mine plans in single or multiple coal seams. This Information Circular presents the improved features of the MULSIM/BM program. The four major improvements are the following: (1) Ability to specify up to 26 material property sets. (2) A new material model capability for gob and inserted materials such as packwalls or cribs. (3) Ability to model a large area with a grid of undivided blocks and to subdivide only a specified fine-mesh portion of these blocks into elements. (4) Specification of extraction ratios for the undivided blocks. In addition, an interactive graphics mesh generator program was written to accompany MULSIM/BM. A complete user's guide and program listings are included within the appendixes. The MULSIM/BM and mesh generator programs are available on tape from the National Technical Information Service.[3]

[1]Mining engineer, Denver Research Center, Bureau of Mines, Denver, CO.

[2]Systems engineer, Electronic Data Systems Corp. (now analyst III, Vanguard Technologies Corp., Denver, CO).

[3]Accession number PB 88-129465. The nine-track computer tape is written at the density of 1,600 bits per inch using unlabeled tape format. The programs are written in ASCII code as 4,960 fixed length records, or card images, with 80 characters per record.

INTRODUCTION

One of the advances occurring in mining technology is the increasing use of computerized mine structural analysis methods to analyze the stability of underground coal mines. Throughout the process of coal extraction from an underground mine, the stresses formerly carried by the newly extracted coal are redistributed to unmined areas, to gob zones, or to artificial structural elements such as packwalls or cribs. This redistribution of stresses can be especially complicated when multiple-seam mining interactions occur. Mine structural analysis methods can calculate the redistribution of overburden stresses and the resulting loadings on any given portion of the underground mine, thus enabling mining engineers to design coal mines that are both safer and yet less wasteful of coal resources.

The Bureau of Mines has conducted research into the application and improvement of state-of-the-art computerized mine structural analysis methods. Theoretical approaches used include finite-element methods, boundary-element methods, and discrete block methods. A subvariation of the boundary-element method called the three-dimensional displacement-discontinuity approach was originally incorporated in a mine structural analysis computer program by Crouch and Fairhurst (3).[4] In this approach, the mine plan for each seam is specified as a grid in plan view, as illustrated in figure 1; the program then calculates the resulting three-dimensional stresses and ground deformations that result from mining. Sinha (15) expanded upon the theory of the approach in three computer programs, one of which, MULSIM, was designed to analyze situations of parallel multiple coal seams. During the 1984-85 period, the Bureau upgraded and completely restructured the MULSIM program into the MULSIM/BM program.

The new MULSIM/BM computer program provides the capability to analyze many coal mining situations and to determine the effects of the three-dimensional stress redistributions caused by mining. The program can analyze problems involving one or more coal seams and can incorporate the effects of previous mining in a large surrounding area, while focusing upon an area of particular interest. Thus, it can be used to analyze the widescale effects of load transfers resulting from previous mining upon the stress levels experienced by particular coal pillars, by unmined coal blocks, or by

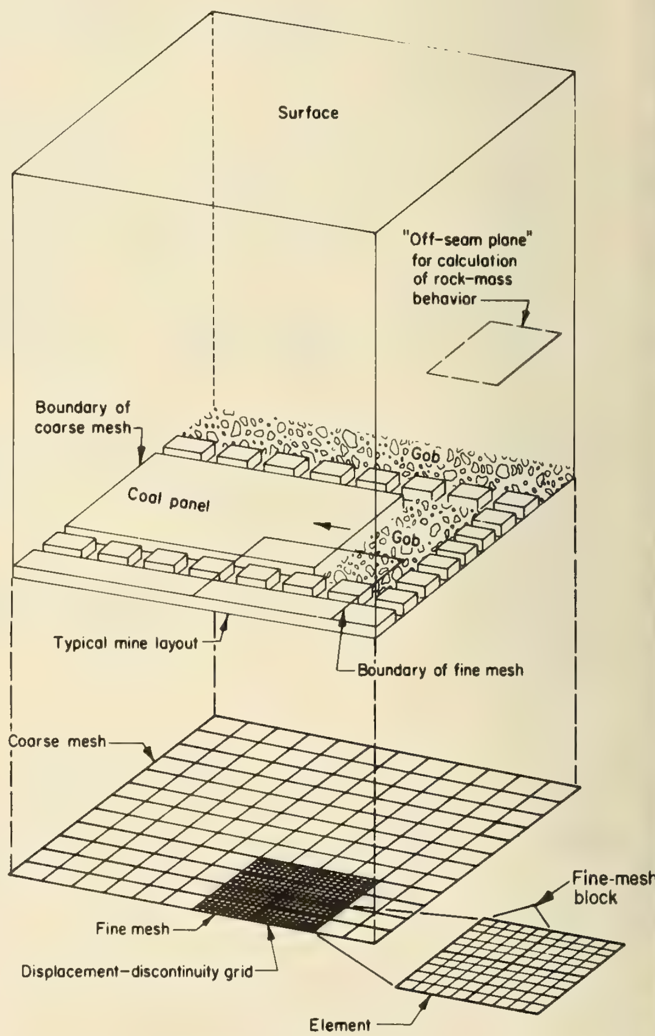


FIGURE 1.—Typical mine layout and displacement-discontinuity grid.

[4]Underlined numbers in parentheses refer to items in the list of references preceding the appendixes.

artificial structural elements such as packwalls, backfill, cribs, or steel frames. Recent Bureau work has shown applications of a preliminary version (9) and of the final version (1) of MULSIM/BM.

Some of the uses envisioned for the MULSIM/BM computer program are as follows: The program can be used to determine the safe size for coal pillars or artificial supports such as packwalls, backfill, cribs, or steel supports used for structural stability. It can be used to predict the magnitude of potential ground-control problems caused by stress concentrations transferred from adjacent mined seams. If the mine plan under consideration is based upon a similar operation elsewhere, or if the in situ stress field or material property conditions are not fully known, the program can conveniently analyze the effects of a different seam dip, different geomechanical properties, and/or a different in situ stress field. Such analyses can assist the mine operator to maximize coal extraction without compromising overall mine safety.

The program is best suited to situations where strata properties are relatively uniform and where localized geological discontinuities such as rolls, kettlebottoms, or faults do not strongly affect the problem. If local discontinuities or property variations are significant, then a finite-element analysis may be a better approach.

This Information Circular documents the improvements incorporated in the MULSIM/BM program and provides a complete user's guide in the appendixes. It also describes a companion mesh generator program for use with MULSIM/BM (fig. 2). For detailed explanation of the mathematical theories incorporated within MULSIM/BM, the reader is referred to the previous literature (3-4, 11, 13-15).

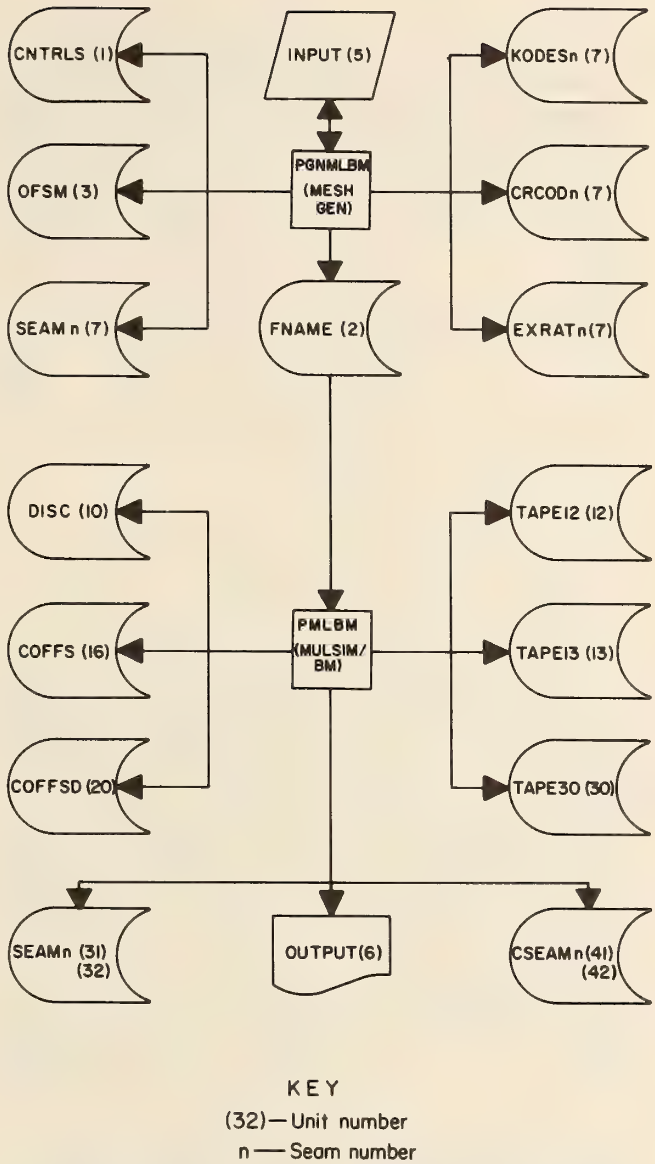


FIGURE 2.—Mesh generator and MULSIM/BM system flowchart.

FEATURES OF MULSIM/BM

The following capabilities of the MULSIM/BM program were carried over from the original MULSIM program (15): MULSIM/BM is designed for analysis of mine designs in single coal seams or in parallel multiple coal seams. The seams can dip at any angle and lie at any desired orientation to the virgin principal stress field. As shown in figure 1, the current mining situation is specified to the program in terms of a grid of elements and blocks; the program then

calculates the resulting seam stresses and closures for each element and block. By specification of "off-seam planes," stresses and rock mass deformations are also available for desired points in the surrounding rock mass.

The rock mass itself is assumed to be a homogeneous, isotropic, infinite elastic body. Thus, since the effect of the surface is omitted, the program is only applicable to problems in which the seam depth is greater than the excavation width being analyzed (15, p. 14). All excavations are assumed to appear instantaneously in a single time step.

New features of MULSIM/BM include

1. Ability to specify up to 26 different material property sets.
2. A new material model capability for inserted materials such as gob, packwalls, or cribs. The inserted material model (or gob model) uses a linear stress-strain relationship with zero stress at zero seam closure.
3. Ability to model a large mining area with a grid of blocks and to subdivide a desired fine-mesh portion of these blocks into elements for greater accuracy.
4. Ability to specify extraction ratios for the undivided or coarse-mesh blocks.

MATERIAL PROPERTY SETS

MULSIM/BM allows definition of as many as 26 different elastic material property sets. A letter of the alphabet is automatically associated with each of the property sets as it is defined. Any of these materials, once defined, can subsequently be used in any portion of any seam plane defined in the model. Each coarse-mesh block and each fine-mesh element defined for an analysis is assigned the desired material property simply by specifying the alphabetic designation as an input for that block or element. This capability is a considerable improvement over the original MULSIM program, in which the analysis was limited to at most one elastic material property set per seam.

To briefly indicate the many possibilities opened up by this capability, consider that in a particular analysis the mining engineer might want to define only six material property sets to include regions of unmined coal, regions of weakened coal in high-stress areas, gob zones, and possibly artifically supported areas such as backfilled openings, packwalls, and openings supported by cribs. Each of these six materials could be used within one or more seam levels. Clearly a very detailed analysis could be conducted using only six materials; and it is therefore unlikely that any analysis would be severely limited by the constraint of having no more than 26 material property sets available.

The input data file is arranged so that a printout of the letter codes for the various blocks and elements corresponds visually to the layout of the grid. This feature makes it very easy to check new data files visually for errors in the layout. It is equally simple to determine the mine layout represented in an old or unfamiliar data file.

INSERTED-MATERIAL OR GOB MODEL

The material behavior capability used to represent fabricated structural components such as gob, packwalls, cribs, steel supports, or backfill is referred to as the inserted-material model. In contrast to the original seam materials, which are in equilibrium with the virgin overburden stresses at zero seam closure (zero roof-to-floor convergence), inserted materials do not support load until some degree of seam closure occurs.

Addition of this capability considerably improved the usefulness of the computer program as applied to practical mine planning. The effect of large gob zones upon the overall load transfer across one or more mined seams can now conveniently be included in an analysis. Likewise the effects of adding artificial support within openings, such as packwalls or cribs, can be analyzed; and the expected loadings upon these supports can be analyzed.

MULSIM/BM requires the program user to input a linear modulus for a gob material

or other inserted material. This modulus is used to calculate the stress carried by the material as seam closure occurs. While this linear modulus may be quite accurate for some materials, such as cribs, for other materials, such as gob, it is only an approximation.

The following subsections provide a detailed discussion of the matters to be considered in selecting an appropriate linear modulus input value for gob. In synopsis, the linear in situ elastic modulus is best viewed as a secant modulus, selected to intercept the true stress-strain curve for gob at a stress level equal to the virgin stress normal to the seam. Because gob stress-strain behavior in situ is not well defined and is very difficult to measure, selection of an appropriate linear in situ elastic modulus value calls for exercise of engineering judgment. In addition, once an in situ elastic gob modulus is selected, the elastic gob modulus input to the computer program must be adjusted to allow for the fact that the program assumes that the gob zone height is only equal to the seam thickness. Since the true in situ gob zone height is necessarily greater than the seam thickness, an adjusted modulus must be used as input to enable the program to calculate the correct gob zone closure values.

The following discussion summarizes some of the mathematical theory relevant to gob modulus selection and also points out some of the considerations that affect the interpretation of results computed using a gob material model.

Theoretical Basis

A mathematical theoretical model of gob zone behavior was first developed by Salamon (14). Mozumdar (11) demonstrated the application of this theoretical model in mine structural analysis using the finite-element method.

The Salamon model of gob behavior is that of a rubblized mass, formed by caving of the immediate roof layers overlying an excavation, that bears no load until it is compacted by the weight of the main, uncaved, rock mass. The load-bearing capacity of the gob increases with compaction, in this mathematical representation, towards an asymptotic limit that represents the volume of the original uncaved rock. The resulting stress-strain relationship is given by Mozumdar (11) as follows:

$$p = k'\omega / (m - \omega) \qquad (1)$$

where p = gob resistence, psi,

k' = a material constant,

m = extracted seam height, in,

and ω = amount of gob zone main-roof-to-floor convergence, in.

Salamon (14) gave a value of 133 psi for the constant, k', as a representative value for European coal measure rocks.

Finite-element analyses conducted by the Bureau using this equation to represent gob behavior have shown that structural analysis using this theory can give useful results for gob behavior. The value of k' given by Salamon is, however, thought to be too low for Western U.S. conditions, and recent Bureau analysis (1, 9) has pointed to a higher value, approximately 1,700 psi, corresponding to use of a secant Young's modulus of 20,000 psi.

To convert equation 1 into a form using stresses and strains, substitute σ_{NL} for p, and divide both the numerator and the denominator by the gob zone height, g, which results in the following equation:

$$\sigma_{NL} = k'\varepsilon_g/(\varepsilon_{max} - \varepsilon_g), \qquad (2)$$

where ε_q = overall normal strain in the gob zone, between the main roof and the floor,

ε_{max} = maximum possible overall normal strain in the gob zone, between the main roof and the floor,

and σ_{NL} = stress carried by the gob zone (nonlinear equation).

Linear In Situ Gob Modulus

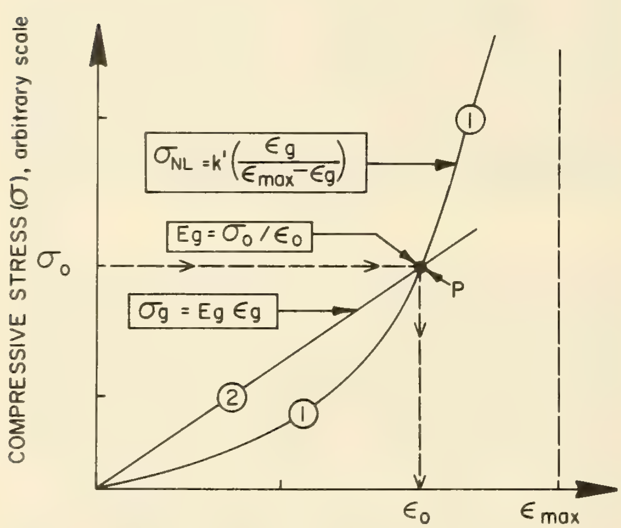
MULSIM/BM requires the program user to input a linear approximation to the nonlinear curve of equations 1 and 2. This linear approximation corresponds to use of a secant elastic (Young's) modulus for the in situ stress-strain response of a gob zone. The linear approximation was chosen to simplify program execution with resulting savings in execution time and also in computer core storage requirements.

Figure 3 shows the recommended method of selecting a secant modulus to correspond to the Salamon curve as applied to a particular virgin overburden stress level. In figure 3, σ_0 is the virgin overburden stress level and ε_{max} is the amount of compressive strain that would correspond to a gob-zone convergence equal to the extracted seam height, m. ε_0 is the strain corresponding to an amount of gob convergence, ω, that just enables the gob to sustain the overburden stress, σ_0. The resulting secant in situ gob modulus, E_g, is then given by

$$E_g = \sigma_0/\varepsilon_0. \qquad (3)$$

When using this method, the extracted seam height, m, must be taken as equal to the full seam thickness, t, that is input to the computer program, since the program assumes that full seam extraction occurs.

In the implementation of this linear gob model in MULSIM/BM, the program begins with a working assumption that all elements of the grid are initially at equilibrium with the virgin overburden stress field. While for an original seam material (coal) this working assumption corresponds to a condition of zero roof-to-floor closure across the seam, for a gob zone it corresponds to an initial closure, c, equal to the product of ε_0 multiplied by the gob zone height, g. This initial working assumption is normally considerably different from the final solution for a particular mining situation.



-1- Nonlinear curve for in situ gob behavior
σ_{NL} Stress sustained by gob (nonlinear equation)
k' Gob material constant
ϵ_g Strain sustained by gob
ϵ_{max} Maximum possible gob strain
P Equilibrium point for gob
-2- Linear approximation
σ_0 Overburden stress
ϵ_0 Equilibrium strain for gob
σ_g Stress sustained by gob (linear equation)
E_g Secant elastic gob modulus (linear modulus)

FIGURE 3.—Selection of a secant elastic modulus for gob.

In the process of iteratively solving for a balanced set of closures, the program normally ends up reducing the closures of gob elements, thus reducing the calculated stress carried by those elements, and increasing the closures of coal elements near the gob zones, thus increasing the stress carried by those elements. In the coal elements, this process results in calculation of stress abutments. In the gob, this process results in calculation of stress-relieved areas near adjacent unmined coal pillars or blocks.

In calculating the precise stresses carried by the gob elements or blocks, the program uses the secant modulus; and in the destressed gob regions the stresses calculated by the secant modulus are slightly higher than the stresses that would be calculated by use of the nonlinear equations 1 and 2. The total stress levels calculated by either method are, however, sufficiently low compared with the overburden stresses that the resulting disparity should in most cases

have little impact on the stresses calculated for other areas of an analysis.

Gob zones are rarely located in such a way that the calculated equilibrium stresses would be greater than the overburden stress. One plausible way for such a condition to occur would be for a gob zone to be located within the influence zone of a remnant pillar in an adjacent seam, which might then cause a considerable stress concentration. Even so, such a condition would rarely cause the calculated loads in a gob zone to exceed the overburden stress unless there were no stiffer load-bearing structures, such as coal pillars or unmined coal blocks, nearby. Because the focus of analytical interest is usually directed towards areas that contain working faces, entries, coal pillars, and the like, the conditions that might cause gob stresses greater than σ_0 were judged unlikely to be of interest; and therefore no provision was made in MULSIM/BM to track the stiffer gob behavior in the region of stresses greater than σ_0.

Recent Bureau analyses have used the MULSIM/BM gob material model successfully to analyze the stress transfers around longwall panel gob zones (1, 9). These analyses have used an assumed in situ elastic gob modulus of 20,000 psi, a number that lies within the middle of a range of 1,000 to 47,000 psi reported by Peng (12, p. 222) for rubblized rock material or gob.

Adjusted Gob Modulus

Reason for Adjustment (1)

The load transfers away from a gob zone can be adequately calculated by the program provided that a properly adjusted gob modulus is used. The total closure of the gob zone corresponds to the flexing of the main rock mass. This rock mass response to mining determines the amount and the lateral extent of the load transfer. A complication arises, however, because the program considers all elements in the grid for a particular seam to have the height of the original seam thickness. In reality, of course, the height of a gob zone is higher: typically, about three to four times the seam height, depending on the bulking factor of the immediate roof that caves to fill the excavation. The resulting discontinuity in height does not per se cause great difficulties, since an underlying assumption of the program is that the element thicknesses are negligible in calculating the response of the rock mass. Because of the discrepancy between the real gob height and the modeled gob height, however, a modified gob modulus must be used in order to calculate the correct gob-zone closures (1, 9).

Derivation of Adjusted Gob Modulus

The derivation of the modified gob modulus is straightforward. For any element, the closure, C, is determined by the equation (1):

$$C = (F/A)/S, \tag{4}$$

where F = the normal force acting upon the element,

A = the area of the element,

and S = the area stiffness of the element (or, stiffness per unit area).

The area stiffness, S, has units of force per cubic inch and is given by the following equation (1):

$$S = M/h, \tag{5}$$

where M = the elastic modulus that is specified as the input to the computer program for the material type used by that element

and h = the element height used by the program (equal to the seam thickness, t).

From equations 4 and 5, calculation of the correct element closure requires maintaining an area stiffness that corresponds to the correct area stiffness of the in situ area represented by that element. It also follows from equation 5

that the in situ effective area stiffness of a gob zone, S_g, can be given by the following equation:

$$S_g = E_g/g, \quad (6)$$

where E_g = the effective in situ elastic (Young's) modulus of the gob

and g = the in situ gob zone height.

As noted, the in situ gob zone height is always greater than the seam thickness, t, and typically is three to four times greater. MULSIM/BM, however, assumes that gob elements also have the height of the original seam thickness, t. Therefore, in order for the program to calculate the correct gob closure, or closure between the main roof and the floor of the gob zone, it is essential to maintain the condition that

$$S = S_g. \quad (7)$$

Thus, the right hand sides of equations 5 and 6 must be maintained equal to each other:

$$M/h = E_g/g. \quad (8)$$

Setting h equal to t and rearranging terms in equation 8 gives the following equation which should be used to calculate the adjusted gob modulus that is needed as input to the computer program (1):

$$M = E_g\ t/g, \quad (9)$$

where E_g = effective in situ elastic (Young's) modulus of the gob,

t = the seam thickness as input to the program,

and g = the in situ gob zone height.

Use of a gob modulus adjusted in this manner has been shown (1, 9) to give results for the overall load transfers that are in reasonable agreement with field experience. As a linear approximation of the in situ modulus of a gob, a number in the range of 1,000 psi to 47,000 psi is consistent with data reported by Peng (12, p. 222). Recent analyses using MULSIM/BM have used an assumed in situ gob modulus of 20,000 psi and an adjusted gob modulus of 5,000 psi (1, 9).

Considerations Affecting Appropriate Use

The following considerations affect the use of the inserted-material model.

Initial Elastic Closure (1)

The program does not account for the elastic closure that occurs in an opening before the new material is inserted, because all excavations are assumed to occur at the same instant; thus, slightly more load may be calculated on the inserted material than would actually occur. For many mining problems, this approximation is acceptable.

For example, if a packwall having an elastic modulus of 25,000 psi is inserted in a 10-ft seam after 1 in of roof closure has occurred, then the additional packwall stress calculated by MULSIM/BM will be approximately 1/120 times 25,000 psi, or about 200 psi of additional stress. The magnitude of this effect may be as much as 5 to 10 pct of the calculated stress in a packwall, a crib, or a gob zone. If the focus of interest is the magnitude of the stress abutment in a pillar or a rib side, then this level of inaccuracy is negligible and is an acceptable tradeoff for the convenient calculation of the overall load transfer effects.

Equivalent Elastic Modulus for Supported Openings (1)

For modeling openings that contain structural supporting members, an adjusted elastic modulus should be used. For example, consider a 20-ft-wide opening supported by steel arches whose legs occupy 3 in of that width, or 1.25 pct of the opening width. This opening is best modeled as an inserted material whose elastic modulus is 1.25 pct of the elastic modulus of the steel in the arches. Another example is the modeling

of packwalls in the gate road system of an advancing longwall. If the packwalls occupy 40 pct of the cross section of the gate road system, then the system can be modeled as an inserted material whose modulus is 40 pct of the effective elastic modulus of the packwalls.

When interpreting the output for an element or block treated in this manner, one must remember that the load carried by the load-bearing member will be averaged across the entire element or block. Thus, the stress output from the program for the two example situations will be as follows: For the 3-in steel legs occupying 1.25 pct of a 20-ft opening width, the normal stress printed out for the element or block will be only 1.25 pct of the actual average normal stress in the steel legs. For the packwalls occupying 40 pct of an opening cross section, the normal stress printed out for the element or block will be 40 pct of the average normal stress in the packwall.

Equivalent Rock Mass Modulus (1)

The program does not account for the disruptive effects of caving upon the rock mass elastic behavior and upon overlying seams. In reality, as the rock mass deflects and descends upon a gob zone, nonelastic effects such as cracking open of joints occur; and the effective modulus of the rock in this region may be considerably lowered. This in situ situation calls for some engineering judgment in the selection of an effective rock mass modulus for the program. If an approximate overall rock mass rating (RMR) can be inferred for the overburden, taking into account that the overburden above a gob zone will be weakened and thus will have a low RMR, then an overall overburden modulus may be estimated from the following equation given by Bieniawski (2):

$$E_M = 2 * \mathrm{RMR} - 100, \qquad (10)$$

where E_M is the overall modulus of deformation of the rock mass, in gigapascals.[5] Recent Bureau works have used values of 400,000 psi (9) and 384,000 psi (1) for the effective in situ rock mass elastic modulus.

[5]1 GPa equals 145,040 psi.

Upper Seams Overlying Gob Zones

The program cannot accurately calculate what would happen to a second seam overlying a gob zone unless the interburden is considerably greater than the expected caving height (1). Major damage will occur to a coal seam overlying a gob zone if the interburden distance is less than a critical distance of 6.3 times the extraction height in the lower seam (6); and if the interburden distance is less than about 3.5 times the lower seam extraction height, the upper seam will be completely destroyed by caving (6).

Therefore, MULSIM/BM should not be used if an overlying coal seam is within 3.5 seam heights of a lower seam gob zone. If the interburden distance is between 3.5 and 6.3 seam heights, the program should be used, if at all, with caution. In such a situation, the elastic properties of coal elements or blocks that overlay the lower seam gob should be reduced according to some estimate of the amount of in situ softening caused by the lower seam extraction. The amount of reduction needed would vary from little or none for 6.3 seam heights of interburden to almost 100 pct reduction for only 3.5 seam heights of interburden.

MODELING LARGE AREAS (1)

The ability to represent a large area is especially useful. In many coal mining situations, the overburden load transfer caused by mining can extend over a very wide area; thus, any given portion of the mine can potentially be affected by extraction at a considerable distance in the same or a neighboring seam.

Nonetheless, the mining engineer often wishes to determine the effect of the total system upon only one or a few of its components, such as a given chain pillar or row of pillars.

MULSIM/BM permits a distinction between coarse-mesh blocks (undivided blocks) and fine-mesh blocks (divided into 25 elements each). The coarse-mesh blocks are

each assigned a material property; and if this is an original seam material (coal), an extraction ratio can also be specified. Coarse-mesh blocks can also be defined as gob or other inserted materials, such as backfill or a packwall system. The area of interest can be included in a fine-mesh area for which the resulting overburden load distributions are calculated on an element-by-element basis.

Coarse-Mesh Blocks

By use of coarse-mesh blocks, the structural characteristics of the mine structure remote from the area of interest can be accurately represented and the resulting effects included in the calculated results; yet the computer calculation time and costs are considerably decreased.

Material Model Definition

Coarse-mesh blocks may use any of the material property sets defined for a particular analysis. The appropriate material property set is specified simply by providing the property set letter, A through Z, as an input associated with the particular coarse block.

Extraction Ratios

For coarse-mesh blocks that represent an original seam material (coal), an extraction ratio is specified in addition to the material property set assignment. This feature allows an analysis to include the effects of previous or current mining activity adjacent to an area of interest without defining all the details of the adjacent mining layout. The program internally reduces the stiffness of a partially extracted coarse-mesh block in accordance with the extraction ratio.

The extraction-ratio feature is appropriate only for mining regions where no roof caving has occurred. If roof caving has occurred in a mined-out region, then this region should be modeled using the gob material model.

Boundary Conditions

Computer programs using the three-dimensional displacement-discontinuity approach normally consider that the region outside the modeled area is at equilibrium with the overburden stress and therefore is implicitly unmined and incompressible. In applications of this method, it has been a matter of concern how best to avoid distortion of results due to the influence of an incompressible model boundary (for example, reference 5, p. 15).

MULSIM/BM deals with this problem fairly easily, through the mechanism of the undivided coarse-mesh blocks. If the coarse-mesh area includes the entire area of interaction that affects the fine-mesh area of interest, as described in the preceding subsection, then the unmodeled regions outside this area of interaction are in a sense irrelevant, since they do not in any case affect the area of interest.

A further refinement of the boundary effect exists in MULSIM/BM because of the potential use of different material property types near the model boundaries. In effect, each individual element or block in the model grid considers that the unmodeled area consists of the same material type that was used for that individual element or block. The result is that any material type, for example a gob zone, can be carried right to the coarse-mesh model boundary. If the blocks close to the boundary are under a stress that is close to the overburden stress, then regardless of material type they will not be significantly affected by the presence of the model boundary.

GOUGE MATERIAL

The MULSIM/BM gouge material capability is taken from the original MULSIM program (15, pp. 140-144). This material model capability allows the use of a seam plane to model a fault plane filled with a gouge material. The resulting capability is very similar to a slip or joint element.

The gouge material uses a Mohr-Coulomb shear strength criterion for calculation of rides (differential horizontal displacements across the material). Besides the Young's modulus and shear modulus, the material model makes use of values of cohesion and internal friction angles. Two sets of these values for cohesion and internal friction angle are used: The first set gives the normal values for an intact material; the second set gives residual values for a material that has cracked open (i.e., with negative closure) and then closed back up again.

In the solution using this material model, the difference between horizontal stresses in the roof and floor of the seam is compared to the maximum allowable shear stresses that are obtained from the operative (intact or residual) strength properties, using the normal compressive stress as a confining stress. If the difference between the horizontal stresses is within the allowable limits, then the rides are determined by using the shear modulus in the normal manner. If, however, the difference between the horizontal stresses exceeds the allowable limits, then the amount of excess stress is used to introduce horizontal slip into the ride components for the next iteration, until the excess horizontal stresses are relieved.

SEAM LIMITATIONS, SIGN CONVENTIONS, AND SYMMETRY OPTIONS

Maximum Number of Seams

In general, the maximum number of seams that can be analyzed depends on the dimension statements within a particular program version being used. The program dimension statements must be fitted to the total computer core memory available, unless the program is to run on a virtual memory machine, in which case the memory limitation effectively disappears.

The original MULSIM program (15) was dimensioned for a maximum of four parallel tabular seams. The MULSIM/BM program, as listed in appendix C, is limited to a maximum of two seams with a maximum grid of 40 by 40 blocks per seam, in order to meet the memory limitations of the computer used for program development. To remain within the same core memory limits, a three-seam maximum would require reducing the maximum grid size to about 30 by 30 blocks per seam, and a four-seam maximum would require a maximum total grid of about 20 by 20 blocks per seam. The mesh-generating program listed in appendix D is compatible with any maximum number of seams and with maximum grid sizes up to 40 by 40 blocks per seam. Appendix G shows how MULSIM/BM can be modified to run on a virtual memory computer.

Location of Seam Origins

As shown in figure 4, the grids for the MULSIM/BM seam planes must be located with the grid origins all lying on the same normal line. This limitation arises

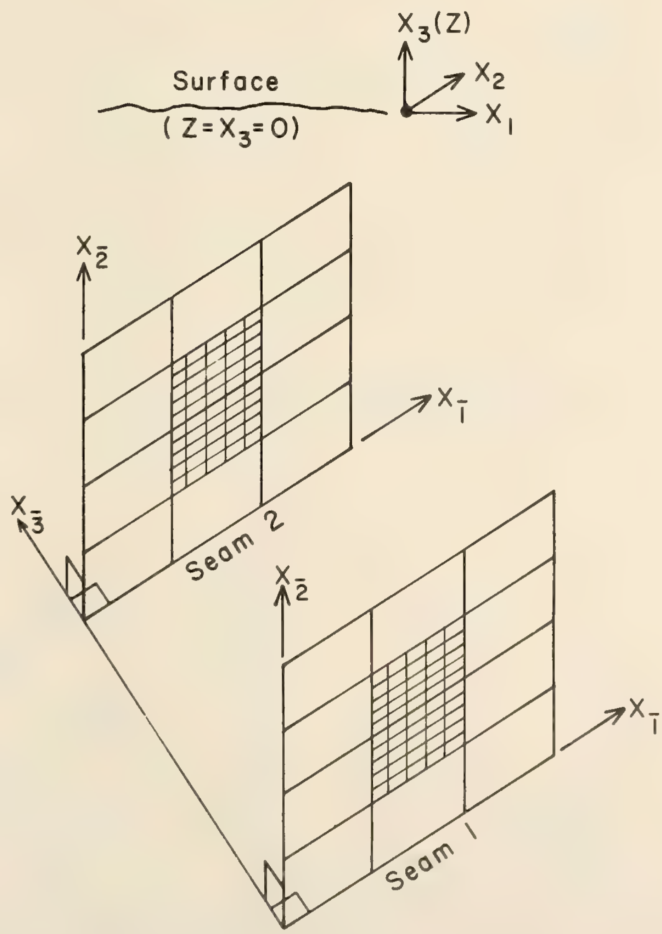


FIGURE 4.—Seam origins lie on the same normal line.

because the programmed calculation of influence coefficients does not allow for relative displacement of the seam grids. Thus, for level seams, the X_1 and X_2 coordinates of all seam origins must be the same. For dipping seams, the X_1, X_2, and X_3 coordinate inputs must be calculated to meet the constraint imposed on the seam origin location.

Sign Conventions

The global coordinate system of MULSIM/BM is a standard set of three orthogonal axes, X_1, X_2, and X_3, that follow the standard right-hand rule: If the thumb of the right hand points in the X_3 direction, then an outstretched finger points in the X_1 direction while a finger curved to 90° points in the X_2 direction. The global origin lies at the surface. For dipping seams, as shown in figure 4, the local $X_{\bar{1}}$ and $X_{\bar{2}}$ axes remain in the plane of the seam, and $X_{\bar{3}}$ is normal to the seam plane.

The sign conventions for displacements and stresses are the same as in the original MULSIM program (15, pp. 33-34) and are shown in figure 5. Positive displacements, μ_1, μ_2, and μ_3, are in the direction of the corresponding global axes, as shown in figure 5*A*. Figure 5*B* shows the positive sense of closure and rides. Seam closure (vertical movement between roof and floor) is positive in the direction that compresses a seam material. Rides (differential horizontal movement between roof and floor) are positive when the floor undergoes a positive relative displacement. Figure 5*C* shows the positive sense of the stress components, σ_{ij}. σ_{ij} has a positive sense if it points in the positive x_j direction while acting on a plane whose

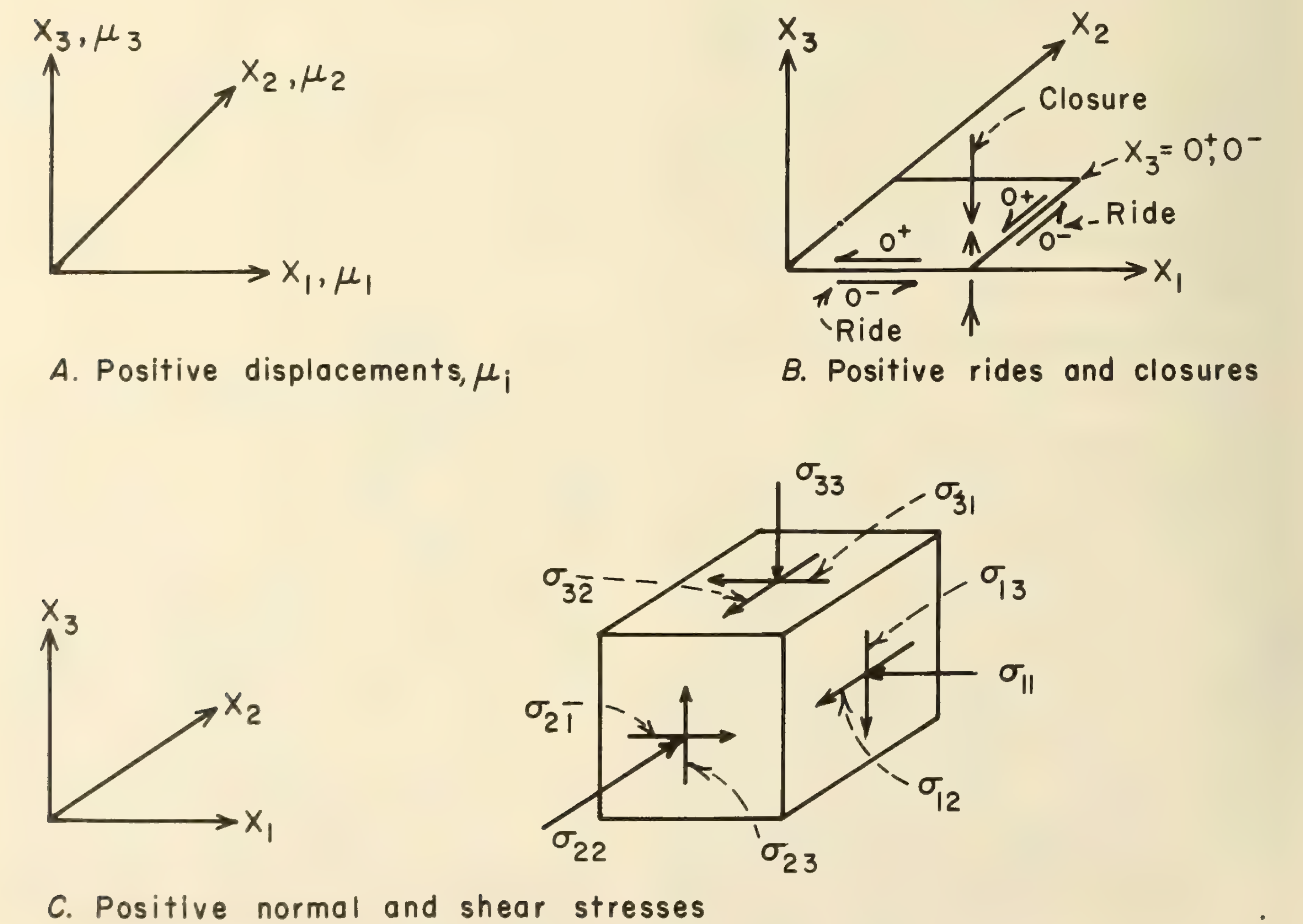


FIGURE 5.—Sign conventions for displacements and stresses.

outward normal points in the negative X_1 direction. Therefore, compressive stresses are positive; and the positive senses of the shear stresses are as shown in figure 5*C*.

Symmetry Options

MULSIM/BM includes the same symmetry options as in the original MULSIM program (15). Figure 6 illustrates the four symmetry options available. As shown in the figure, option 1 specifies no symmetry. Options 2 and 3 allow a line of symmetry through the center of the model grid, and option 4 combines options 2 and 3 so that two lines of symmetry exist.

The effect of specifying symmetry option 2, 3, or 4 is to reduce the solution time and, to a small extent, the input data requirements. Only the nonsymmetrical portion of the grid is specified as

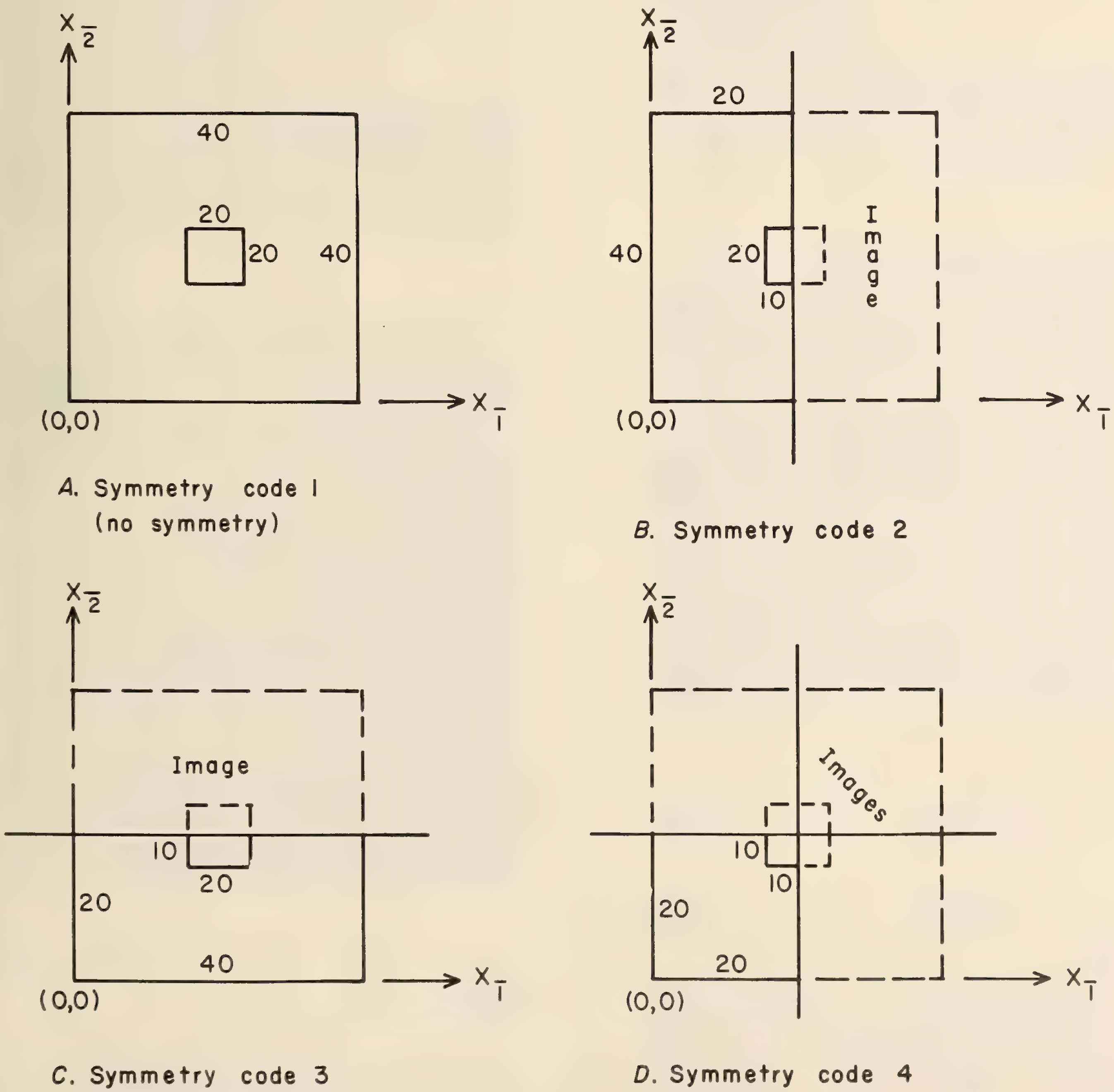


FIGURE 6.—Symmetry options.

input. For example, for the program as listed in appendix C, the maximum grid size is 40 by 40 blocks, including a maximum 20- by 20-block fine-mesh grid; but if symmetry code 4 is used, only a 20- by 20-block grid with a maximum 10- by 10-block fine-mesh grid can be input. The fine-mesh area must be located so that the lines of symmetry also bisect the fine-mesh grid. For symmetry codes 2 and 3, the fine-mesh area can lie anywhere along the line of symmetry, so long as the line of symmetry bisects the fine-mesh grid; for symmetry code 4, the fine-mesh grid must necessarily occupy the center of the coarse-mesh grid.

The savings in solution time occurs because the program needs only to find solutions for the nonsymmetric portion (half or quadrant) of the grid. During the iteration process, the current solutions for each element and block are also stored for the corresponding imaged elements and blocks. Total solution time in the iteration phase is thus reduced by a factor of either 2 or 4.

COMPANION MESH GENERATOR PROGRAM

The companion interactive-graphics mesh generator program is quick and convenient. The different material types are each assigned elastic property values (Young's modulus and shear modulus) and an identifying letter (A through Z); each element or undivided block is then assigned the desired material type by specifying the appropriate identifying letter. For the undivided coarse-mesh blocks, a table of extraction ratios can be defined; each undivided block that uses an original seam material can then be assigned one of the extraction ratios.

Figure 7 shows an example mining layout that was defined to illustrate use of MULSIM/BM and of the companion mesh-generator program. Example grids for analysis of this layout are shown in figure 8. A mining engineer can easily generate a data file to analyze this example problem in a 1-h session using the mesh-generator program.

Appendix B shows a sample terminal session using the mesh generator to create a MULSIM/BM input file for this example. Appendix E shows the resulting MULSIM/BM input data file. Appendix F shows the output that results from MULSIM/BM for the same example. The job control language needed to run the program on the computer system used at the Bureau's Denver (CO) Research Center is shown in appendix H.

DEVELOPING A PLAN OF ANALYSIS

The following procedure is recommended as a general approach to analyzing a problem using MULSIM/BM. The procedure presented here expands upon a general approach recommended previously (1). While some of the recommended steps may appear obvious, it is worth taking note that none of them should, in general, be omitted.

1. Define the problem. This means explicitly asking one or more questions that one expects can be answered on the basis of analysis. Examples of such questions might be, "What is the safe size for a particular coal pillar?" or "How much steel support is needed in a certain area to limit opening closure to desired limits?"

2. Identify the appropriate theoretical approach to analyze the problem. For some problems a finite-element approach or perhaps a discrete block analysis will be indicated. This selection of approach must be made on the basis of the analyst's experience and familiarity with the capabilities and limitations of different available methods. The following steps assume that MULSIM/BM has been selected as the vehicle of the analysis.

3. Define the area of interest. This area should be included within the fine-mesh zone of the grid.

4. Identify the relevant area of interaction. This will be the maximum area from which load transfer effects will

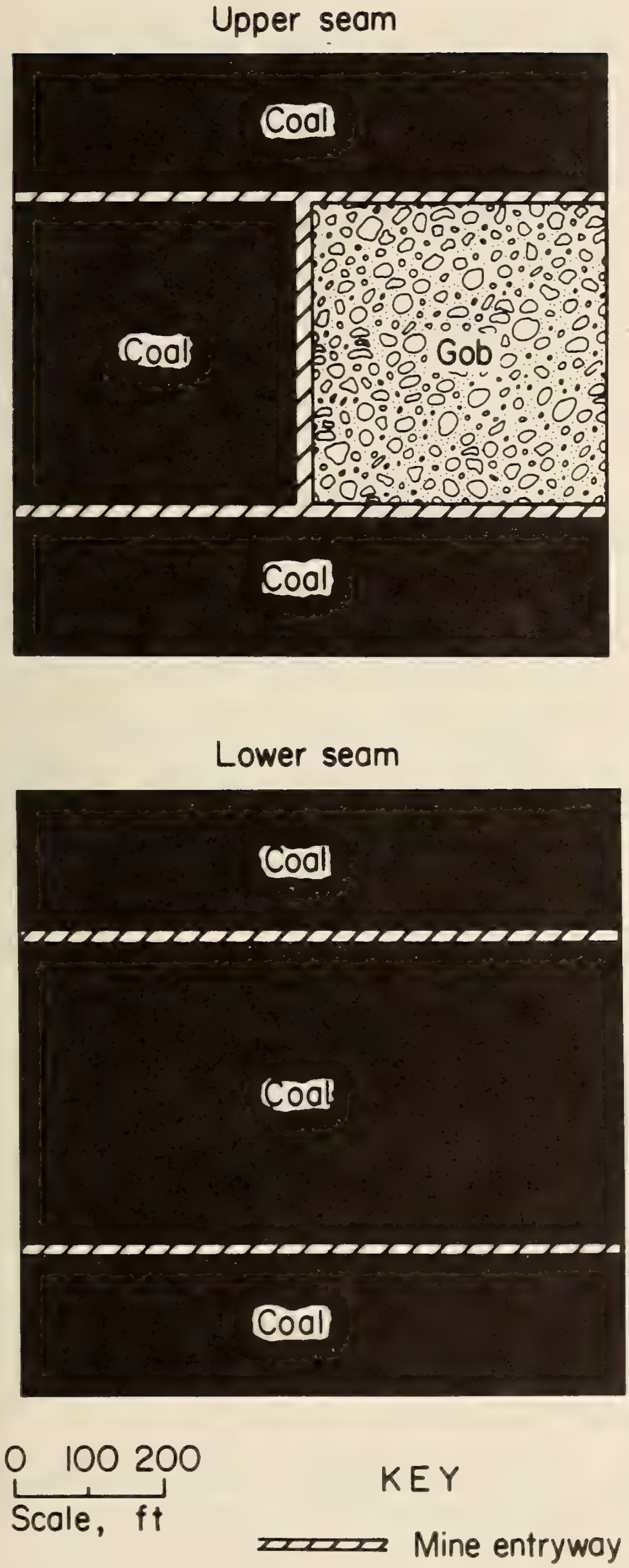


FIGURE 7.—Example mining layout.

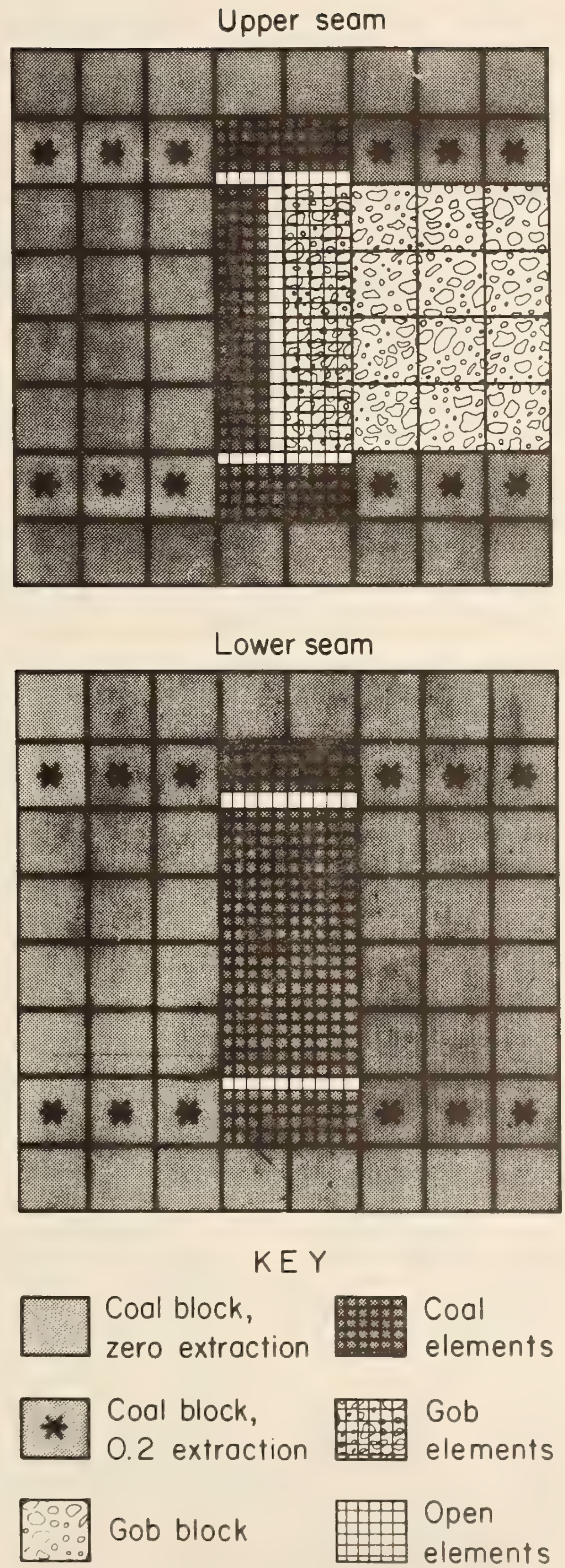


FIGURE 8.—MULSIM/BM grids for the example layout.

be felt within the area of interest. If possible, this entire area should be included within the MULSIM/BM grid.

5. Identify reasonable initial values for effective material properties and for the virgin stress conditions. True stress values can only be determined by in situ measurements. Material properties are sometimes available from laboratory or in situ testing; in other cases, representative properties can be used with reasonable accuracy. As discussed previously in this report, recent Bureau work (1, 9) has used rock mass modulus values in the range of 400,000 psi, corresponding to relatively low values of the rock mass rating (2), and an effective in situ gob modulus of 20,000 psi, adjusted to 5,000 psi in accordance with equation 9 of this report.

The initial values used for the coal seam or for other materials such as packwalls may need to be reduced in highly stressed areas such as those adjacent to a large gob zone or within the influence zone of a stress concentration emanating from an adjacent seam. Previous Bureau work (1, 7-10) has shown that a procedure of reducing the effective modulus value until the resulting stresses become allowable can adequately represent the transfers of stresses that result from in situ weakening and softening of highly stressed materials.

6. Identify cases to be analyzed.

7. Generate MULSIM/BM input files for the desired cases using the mesh generator program.

8. Run MULSIM/BM to obtain results for the desired cases.

9. Analyze the results using postprocessing graphics or other methods.

10. Determine whether the problem has been satisfactorily solved. If not, it may frequently be desirable to redefine the problem on the basis of the analysis conducted so far. At this stage, all the parameters of the analysis should be reexamined; even the selection of theoretical approach may be subject to change. Therefore the analyst should at this point return to step 1 of this suggested analysis procedure.

REFERENCES

1. Beckett, L. A., and R. S. Madrid. Practical Application of MULSIM/BM for Improved Mine Design. Paper in Use of Computers in the Coal Industry (Third Conf., WV Univ., Morgantown, WV, July 28-30, 1986). A. A. Balkema, Rotterdam, Netherlands, 1986, pp. 209-219.

2. Bieniawski, Z. T. Determining Rock Mass Deformability: Experience From Case Histories. Int. J. Rock Mech., Min. Sci. & Geomech. Abstr., v. 15, 1978, pp. 237-247.

3. Crouch, S. L., and C. Fairhurst. The Mechanics of Coal Mine Bumps and the Interaction Between Coal Pillars, Mine Roof, and Floor (contract H0101778, Univ. MN). BuMines OFR 53-73, 1973, 88 pp.; NTIS PB 222 898.

4. Crouch, S. L., and A. M. Starfield. Boundary Element Methods in Solid Mechanics. Allen & Unwin, 1983, 322 pp.

5. Hardy, M. P., H. N. Maleki, and S. J. Mitchell. Application of Displacement of Discontinuity Methods in Underground Mine Design. Paper in Rock Masses: Modeling of Underground Openings/Probability of Slope Failure/Fracture of Intact Rock, ed. by C. H. Dowding (Proc. Symp. sponsored by ASCE Geotech. Eng. Div. in conjunction with ASCE Conv., Denver, CO, Apr. 29-30, 1985). ASCE, 1985, pp. 14-20.

6. Hladysz, Z. Analysis of Risk in Multiple Seam Mining. Soc. Min. Eng. AIME preprint 85-357, 11 pp.

7. Kidybinski, A., and C. O. Babcock. Stress Distribution and Rock Fracture Zones in the Roof of Longwall Face in a Coal Mine. Rock Mech., v. 5, No. 1, 1973, pp. 1-19.

8. Kripakov, N. P. Analysis of Pillar Stability on Steeply Pitching Seam Using the Finite Element Method. BuMines RI 8579, 1981, 33 pp.

9. Kripakov, N. P., L. A. Beckett, and D. A. Donato. Loading on Underground Mine Structures Influenced by Multiple Seam Interaction. Paper in International Symposium on Application of Rock Characterization Techniques in Mine Design (SME-AIME Ann. Meeting, New Orleans, LA, Mar. 1986). Soc. Min. Eng. AIME, Littleton, CO, 1986, pp. 225-236.

10. Kripakov, N. P., and M. T. Melvin. A Computer Program To Simulate Progressive Rock Failure Around Coal Mine Entries. Paper in Proceedings of the First Conference on Use of Computers in the Coal Industry (WV Univ., Morgantown, WV, Aug. 1-3, 1983). Soc. Min. Eng. AIME, New York, 1983, pp. 487-502.

11. Mozumdar, B. J. A Mathematical Model of Ground Movement Due to Underground Mining. Ph.D. Thesis, PA State Univ., 1974, 134 pp. Xerox Univ. Microfilms, Ann Arbor, MI, fac. 75-9815, 1978.

12. Peng, S. S. Coal Mine Ground Control. Wiley, 1978, 450 pp.

13. Salamon, M. D. G. Elastic Analysis of Displacements and Stresses Induced by the Mining of Seam or Reef Deposits. J. South African Inst. Min. & Metall., Part I, v. 64, No. 4, 1963, pp. 128-149; Part II, v. 64, No. 6, 1964, pp. 197-218; Part III, v. 64, No. 10, 1964, pp. 468-500; Part IV, v. 65, No. 5, 1964, pp. 319-338.

14. _____. Reconsolidation of Caved Areas. Coll. Res. Lab. Transvaal and Orange Free State Chamber of Mines Res. Org., Republic of South Africa, Res. Rep. 58/66, July 1966, 25 pp.

15. Sinha, K. P. Displacement Discontinuity Technique for Analyzing Stresses and Displacements Due to Mining in Seam Deposits. Ph.D. Thesis, Univ. MN, 1979, 311 pp.; Univ. Microfilms Int., Ann Arbor, MI, fac. 7918390, 1982.

APPENDIX A.--MULSIM/BM USER'S GUIDE

This appendix explains to the user of MULSIM/BM how to create the input data file.

The coarse mesh features and the method of assigning material properties to elements are new within this program. Many of the remaining data definition requirements are the same as the original MULSIM program as documented by Sinha (15).[1]

The input data file may be created by using any available text editor, or by running the companion mesh generator program. If the user's computer system has Tektronix graphics software Plot-10, and the appropriate terminals (Tektronix 4010, 4014, etc.), then the best way to generate the input data file is to use the mesh generator program.

The mesh generator program is described fully in appendix B.

The maximum grid size and maximum number of seams allowed depend on the particular MULSIM/BM version. The version listed in appendix D allows a maximum grid of 40 by 40 blocks and a maximum of two seams.

---

[1] Underlined numbers in parentheses refer to items in the list of references preceding this appendix.

INPUT DATA INSTRUCTIONS

THE INPUT FILE CONSISTS OF SEVERAL DIFFERENT CARD TYPES. THE FILE MUST BE IN CARD TYPE SEQUENCE.

THE INPUT DATA MUST BE IN CONSISTENT UNITS FOR DIMENSIONS, STRESSES, AND MODULI.
FOR EXAMPLE:

METERS	KILOPASCALS	KILOPASCALS
METERS	MEGAPASCALS	MEGAPASCALS

CARD TYPE 1 - TITLE

DATUM	COLUMNS	FORMAT
TITLE (ANY TEXT)	1-80	20A4

CARD TYPE 2 - HOST ROCK MATERIAL PROPERTIES

DATUM	COLUMNS	FORMAT
POISSON'S RATIO OF ROCK MASS	1-8	F8.2
MODULUS OF ELASTICITY OF ROCK MASS	9-20	E12.6
NUMBER OF SEAMS(1-N)	21-28	I8

N - THE CURRENT VERSION WILL ACCEPT UP TO 2 SEAMS. THIS IS BECAUSE OF THE MAXIMUM 40X40 BLOCK MODEL. OTHER VERSIONS WITH SMALLER BLOCK MODELS WILL ACCEPT UP TO 4 SEAMS.

CARD TYPE 3 - NUMBER OF MATERIAL PROPERTIES

DATUM	COLUMNS	FORMAT[2]
NUMBER OF MATERIAL TYPES(1-26)	1-80	*

CARD TYPE 4 - SEAM MATERIAL PROPERTIES
(ONE CARD FOR EACH MATERIAL TYPE INPUT IN CARD TYPE 3)

DATUM	COLUMNS	FORMAT
MODULUS OF ELASTICITY	1-16	E16.10
SHEAR MODULUS (MODULUS OF RIGIDITY)	17-32	E16.10

THE FOLLOWING DATA MAY BE IGNORED IF THE SHEAR FAILURE OF SEAM MATERIAL IS NOT TO BE CONSIDERED.(STATUS CODE IS 3)

COHESION	33-40	E8.2
FRICTION ANGLE	41-48	E8.2
RESIDUAL VALUE OF COHESION	49-56	E8.2
RESIDUAL VALUE OF FRICTION ANGLE	57-64	E8.2

[2] AN '*' FORMAT IS FREE FORMAT(NUMBERS SEPARATED BY COMMAS)

GOB MATERIAL INDICATOR(1=YES,0=NO)	65-72	I8

CARD TYPE 5 - PRIMITIVE STRESS PARAMETERS

THE STRESS TENSOR THAT DESCRIBES A STRESS STATE THAT EXISTS PRIOR TO MINING IS DEFINED AS LINEAR FUNCTIONS OF GLOBAL DIRECTION. WITH COMPRESSION BEING POSITIVE; THE EQUATIONS ARE OF THE FORM:

STRESS(1,1) = A11 - B11 * X3
(1,2) = A12 - B12 * X3 AND SO ON.
AXX MUST BE >= -.9999 AND <= 99999.
BXX MUST BE >= -.9999 AND <= 9.9999
X3 MUST BE NEGATIVE.

REFER TO SINHA (15) FOR MORE DETAILS.

DATUM	COLUMNS	FORMAT
A11 CONSTANT	1-6	F6.0
B11 CONSTANT	7-12	F6.4
A12 CONSTANT	13-18	F6.0
B12 CONSTANT	19-24	F6.4
A13 CONSTANT	25-30	F6.0
B13 CONSTANT	31-36	F6.4
A22 CONSTANT	37-42	F6.0
B22 CONSTANT	43-48	F6.4
A23 CONSTANT	49-54	F6.0
B23 CONSTANT	55-60	F6.4
A33 CONSTANT	61-66	F6.0
B33 CONSTANT	67-72	F6.4

CARD TYPE 6 - MODEL DATA

DATUM	COLUMNS	FORMAT
BLOCK WIDTH	1-8	F8.2
NUMBER OF BLOCKS IN X-AXIS(1 TO 40)	9-16	I8

THIS IS THE NUMBER OF BLOCKS ALONG THE LOCAL X-AXIS IN EACH SEAM. (TOTAL COARSE GRID PLUS FINE GRID). IF SYMMETRY IS SPECIFIED, THEN THIS SHOULD BE THE UNIQUE PORTION(QUADRANT OR HALF) OF THE MESH.

NUMBER OF BLOCKS IN Y-AXIS(1 TO 40)	17-24	I8

SAME AS ABOVE EXCEPT FOR THE Y-AXIS.

SYMMETRY FLAG(1,2,3,4)	25-32	I8

IF SYMMETRY IS SPECIFIED, THE LINES OF SYMMETRY PASS THROUGH THE CENTER OF THE MODEL.

1 - NO SYMMETRY
2 - VERTICAL LINE OF SYMMETRY
3 - HORIZONTAL LINE OF SYMMETRY
4 - VERTICAL AND HORIZONTAL LINES OF SYMMETRY

NUMBER OF OFF-SEAM PLANES AT WHICH DISPLACEMENTS AND STRESSES ARE TO BE COMPUTED.	33-40	I8

THE FINE MESH BLOCKS ARE SUBDIVIDED INTO 25 ELEMENTS. THIS CONSISTS OF 5 ROWS OF 5 ELEMENTS PER ROW.

STARTING BLOCK IN FINE MESH X-AXIS	41-48	I8
ENDING BLOCK IN FINE MESH X-AXIS	49-56	I8
STARTING BLOCK IN FINE MESH Y-AXIS	57-64	I8
ENDING BLOCK IN FINE MESH Y-AXIS	65-72	I8

NOTE: ENDING BLOCK - STARTING BLOCK MUST BE LESS THAN 20

CARD TYPE 7 - EXTRACTION DATA FOR COARSE MESH BLOCKS

EXTRACTION CODE '0' IS THE DEFAULT FOR ZERO EXTRACTION. THIS CODE IS IRRELEVANT FOR GOB OR OPENINGS.

DATUM	COLUMNS	FORMAT
NUMBER OF EXTRACTION RATIOS NUMBER OF ADDITIONAL EXTRACTION RATIO CODES.(ADDITIONAL TO ZERO). VALID VALUES ARE 1 THROUGH 9.	1-8	I8
EXTRACTION RATIO 1 ALL EXTRACTION RATIOS MUST BE A FRACTIONAL NUMBER GREATER THAN 0 AND LESS THAN 1.	9-16	F8.3
EXTRACTION RATIO 2	17-24	F8.3
EXTRACTION RATIO 3	25-32	F8.3
EXTRACTION RATIO 4	33-40	F8.3
EXTRACTION RATIO 5	41-48	F8.3
EXTRACTION RATIO 6	49-56	F8.3
EXTRACTION RATIO 7	57-64	F8.3
EXTRACTION RATIO 8	65-72	F8.3
EXTRACTION RATIO 9	73-80	F8.3

CARD TYPE 8 - GLOBAL COORDINATES OF GRID ORIGIN
(ONE CARD FOR EACH SEAM AS INPUT IN CARD TYPE 2)
(THE SEAM ORIGINS ARE AT THE ROOF OF THE SEAM)

DATUM	COLUMNS	FORMAT
X-COORDINATE FOR SEAM ORIGIN	1-8	F8.1
Y-COORDINATE FOR SEAM ORIGIN	9-16	F8.1
Z-COORDINATE FOR SEAM ORIGIN(MUST BE NEGATIVE)	17-24	F8.1
SEAM THICKNESS	25-32	F8.1

CARD TYPE 9 - GRID ORIENTATION FOR SEAM
(DIRECTION COSINES BETWEEN SEAM GRID AND GLOBAL COORDINATE SYSTEM)
EN(I,J) IS THE DIRECTION COSINE OF LOCAL AXIS X(I) AT THE SEAMS WITH RESPECT TO THE GLOBAL AXIS X(J).

DATUM	COLUMNS	FORMAT
EN(1,1)	1-8	F8.5
EN(2,1)	9-16	F8.5
EN(3,1)	17-24	F8.5
EN(1,2)	25-32	F8.5
EN(2,2)	33-40	F8.5
EN(3,2)	41-48	F8.5
EN(1,3)	49-56	F8.5
EN(2,3)	57-64	F8.5
EN(3,3)	65-72	F8.5

CARD TYPE 10 - MODEL RUN AND RESTART CONTROL PARAMETERS

DATUM	COLUMNS	FORMAT
OVER RELAXATION FACTOR VALID VALUES 1.0 THROUGH 1.99 (1.35 IS OFTEN BEST)	1-8	F8.2
MAXIMUM ITERATIONS FOR THIS RUN	9-16	I8
COMPUTER RUN NUMBER (SEE NOTE 1.)	17-24	I8
COMPLETED ITERATIONS(SEE NOTE 2.) IN PREVIOUS RUNS	25-32	I8
USE PREVIOUS GRID FLAG(1=Y,0=N)(SEE NOTE 3.)	33-40	I8

NOTES:
1. COMPUTER RUN NUMBER GREATER THAN 1 IMPLIES THAT DESIRED ACCURACY WAS NOT ACHIEVED IN PREVIOUS RUN.
2. COMPLETED ITERATIONS GREATER THAN ZERO IMPLIES THAT THE RIDE AND CLOSURE VALUES OBTAINED AND SAVED IN PREVIOUS RUNS ARE TO BE USED AS THE INITIAL APPROXIMATIONS FOR THIS RUN.
3. IF THE PREVIOUS GRID FLAG IS SET TO '1', THEN THE INFLUENCE COEFFICIENTS ARE TO BE RETRIEVED FROM A PREVIOUSLY SAVED FILE. ALSO THE MODEL GRID IS TO REMAIN THE SAME AS IN A PREVIOUS RUN.

CARD TYPE 11 - FINE MESH MATERIAL PROPERTY CODES
(ONE CARD FOR EACH ROW OF ELEMENTS IN THE FINE MESH GRID OF EACH SEAM)

EACH CARD CONTAINS THE MATERIAL PROPERTY CODES FOR ALL ELEMENTS IN THE ROW.
(THE NUMBER OF ELEMENTS PER ROW IS 5 TIMES THE NUMBER OF FINE MESH BLOCKS IN THE X-DIRECTION. THE TOTAL NUMBER OF ELEMENT ROWS IS 5 TIMES THE NUMBER OF FINE MESH BLOCKS IN THE Y-DIRECTION, TIMES THE NUMBER OF SEAMS.)

DATUM	COLUMNS	FORMAT
MATERIAL PROPERTY CODES	1-100	100A1

EACH CODE ENTERED SPECIFIES THE MATERIAL
PROPERTY SET FOR A PARTICULAR ELEMENT.
THE CODES MUST CORRESPOND TO THE NUMBER OF
MATERIAL TYPES ENTERED IN CARD TYPE 3, PLUS THE VALUES 1 AND 2.
I. E. IF 2 MATERIAL TYPES ENTERED, THEN
VALID VALUES ARE (A,B, 1, 2).
IF 3 MATERIAL TYPES ENTERED, THEN
VALID VALUES ARE (A,B,C, 1, 2) ETC.
CODE 1 INDICATES AN OPENING, CODE 2 INDICATES A COMPLETELY RIGID MATERIAL.

CARD TYPE 12 - FINE MESH MINING STATUS CODES
(ONE CARD FOR EACH ROW OF ELEMENTS IN THE FINE MESH GRID
OF EACH SEAM)

EACH CARD CONTAINS THE MINING STATUS CODES FOR
ALL ELEMENTS IN THE ROW.
(THE NUMBER OF ELEMENTS PER ROW IS 5 TIMES THE NUMBER OF
FINE MESH BLOCKS IN THE X-DIRECTION. THE TOTAL NUMBER
OF ELEMENT ROWS IS 5 TIMES THE NUMBER OF FINE MESH BLOCKS
IN THE Y-DIRECTION TIMES THE NUMBER OF SEAMS.)

DATUM	COLUMNS	FORMAT
MINING STATUS CODES	1-100	100A1

EACH CODE ENTERED SPECIFIES THE
MINING STATUS CODE SET FORA
PARTICULAR ELEMENT.
VALID VALUES ARE:
3 - NORMAL PROPERTIES
(SEAM MATERIAL,GOB
INSERTED MATERIAL)
4 - GOUGE MATERIAL
(MOHR-COULOMB MODEL)

CARD TYPES 13 AND 14 MUST BE INCLUDED IF A COARSE MESH AREA HAS BEEN DEFINED. THEY MUST NOT BE INCLUDED IF THE MODEL DOES NOT CONTAIN ANY COARSE MESH BLOCKS.

CARD TYPE 13 - COARSE MESH MATERIAL PROPERTY CODES
(ONE CARD FOR EACH ROW OF BLOCKS IN THE TOTAL GRID MODEL)
(EACH CARD CONTAINS THE MATERIAL PROPERTY CODES FOR ALL
BLOCKS IN THE ROW)

DATUM	COLUMNS	FORMAT
MATERIAL PROPERTY CODES	1-40	40A1

THE VALID CODES ARE THE SAME AS THE
FINE MESH AREA WITH THE ADDITIONAL
SPECIAL CODE '*' WHICH IS USED TO
DESIGNATE A FINE MESH BLOCK.

CARD TYPE 14 - COARSE MESH EXTRACTION RATIO CODES

(ONE CARD FOR EACH ROW OF BLOCKS IN THE TOTAL GRID MODEL)
EACH CARD CONTAINS THE EXTRACTION CODE FOR EVERY BLOCK IN THE ROW.

DATUM	COLUMNS	FORMAT
EXTRACTION RATIO CODES	1-40	40A1

EACH CODE ENTERED SPECIFIES THE EXTRACTION RATIO FOR A PARTICULAR BLOCK IN THE ROW. VALID CODES (1 TO 9) ARE SET UP IN CARD TYPE 7.
I.E. IF 2 ENTERED, THEN THE VALID VALUES ARE (0,1,2).
IF 3 ENTERED, THEN THE VALID VALUES ARE (0,1,2,3) ETC.
AGAIN, THE SPECIAL CODE '*' IS USED TO DESIGNATE A FINE MESH BLOCK.

CARD TYPE 15 - OFF SEAM GRID DETAILS

THE OFF-SEAM PLANE IS AN ARBITRARY PLANE IN THE ROCK MASS. STRESSES AND DISPLACEMENTS ARE CALCULATED AT THE CENTER OF EACH BLOCK WITHIN THE OFF-SEAM PLANE.

DATUM	COLUMNS	FORMAT
X-COORDINATE OF OFF-SEAM ORIGIN	1-12	F12.3
Y-COORDINATE OF OFF-SEAM ORIGIN	13-24	F12.3
Z-COORDINATE OF OFF-SEAM ORIGIN	25-36	F12.3
BLOCK WIDTH ALONG X-AXIS	37-48	F12.3
BLOCK WIDTH ALONG Y-AXIS	49-60	F12.3
NUMBER OF BLOCKS ALONG X-AXIS	61-68	I8
NUMBER OF BLOCKS ALONG Y-AXIS	69-76	I8

CARD TYPE 15A - OFF SEAM PLANE ORIENTATION

(DIRECTION COSINES)

DATUM	COLUMNS	FORMAT
EN1X	1-8	F8.5
EN2X	9-16	F8.5
EN3X	17-24	F8.5
EN1Y	25-32	F8.5
EN2Y	33-40	F8.5
EN3Y	41-48	F8.5
EN1Z	49-56	F8.5
EN2Z	57-64	F8.5
EN3Z	65-72	F8.5

NOTE: CARD TYPE PAIRS(15,15A) ARE REPEATED FOR EACH OFF-SEAM PLANE

APPENDIX B.--MULSIM/BM MESH GENERATOR USER'S GUIDE

This user's guide is in the form of a sample terminal session. The MULSIM/BM input file created by this sample session is listed in appendix E.

As installed on the computer system used by the Bureau, the mesh generator is started by entering the following commands following the "ready" prompt from the computer:

```
GET,PRGENM/UN=BM0001L
BEGIN,PRGENM
```

These commands will start up a procedure file that calls the mesh generator program. A listing of the procedure file follows the sample terminal session. Anyone using a different computer system will need to implement an equivalent procedure.

The user will make the selection as to whether or not the mesh generator will convert depth and distance entered in feet to inches.

All card type data will be requested as outlined in the input data instructions in appendix A, except for the orientation of the grid or the orientation of any off-seam planes. The orientation will be supplied in terms of strike, dip, and offset angles, and the mesh generator program will then calculate the appropriate direction cosines.

The maximum grid size and maximum number of seams depend on the MULSIM/BM version to be used.

SAMPLE TERMINAL SESSION

```
 READY
 GET,PRGENM/UN=BM0001L                         USER
 BEGIN,PRGENM                                  USER
 THIS PROGRAM GENERATES INPUT-DATA FOR MULSIM

 WHAT DO YOU WANT TO DO NOW
     TYPE 0---TO END THIS PROGRAM NOW
     TYPE 1---TO GENERATE A DATA FILE
? 1                                                                        USER
 PREPARATION OF INPUT-DATA FOR --ENHANCED MULSIM---
  YOU MUST USE CONSISTENT UNITS FOR:
           DIMENSION,    STRESSES,  AND      MODULI
  (FOR EXAMPLE: METERS, KILOPASCALS,       KILOPASCALS
                METERS, MEGAPASCALS,       MEGAPASCALS
                INCHES,         PSI,               PSI
   DO YOU WANT AUTOMATIC CONVERSION OF DEPTH AND
   DISTANCE ENTERED IN FEET TO INCHES(Y OR N)
 ? Y                                                                       USER
  DEPTH AND DISTANCE VALUES WILL BE CONVERTED
  FROM FEET TO INCHES.
  (MODULUS VALUES WILL NOT BE CONVERTED)
 CARD TYPE 1--TITLE
  WHAT IS THE TITLE
 ? MULSIM/BM SAMPLE PROBLEM -- THIS IS LINE 1 OF THE FILE     USER
 TITLE ENTERED IS:
MULSIM/BM SAMPLE PROBLEM -- THIS IS LINE 1 OF THE FILE
 IS THIS OK(Y/N)
 ? Y                                                                       USER
 CARD TYPE 2---
   POISSON'S RATIO OF THE ROCK MASS =
 ? .15                                                                     USER
MODULUS OF ELASTICITY OF THE ROCK MASS =
 ? 400000                                                                  USER
THE MAXIMUM GRID AND THE MAXIMUM
NUMBER OF SEAMS ALLOWED DEPEND
ON THE MULSIM/BM VERSION TO BE USED
     1 - 2           40 X 40
         3           30 X 30
         4           20 X 20
NUMBER OF SEAMS =
 ? 2                                                                       USER
HOST MATERIAL PROPERTIES ENTERED ARE:
POISSON'S RATIO: .15
MODULUS OF ELASTICITY: 400000.
NUMBER OF SEAMS: 2
 IS THIS OK(Y/N)
 ? Y                                                                       USER
```

```
  CARD TYPE 3---
   SEAM MATERIALS ARE SPECIFIED BY A LETTER CODE
    (A,B,C,....Z)
   HOW MANY OF THESE MATERIALS ARE THERE
? 2                                                                    USER
 CARD TYPE 4
  MATERIAL NUMBER: ---1
  MATERIAL CODE:    ---A
  IS THIS A
   GOUGE MATERIAL(MOHR-COULOMB MODEL)?
  3 = "NORMAL MATERIALS"
     (SEAM MATERIALS,GOB, OR "INSERTED" MATERIALS)
  4 = GOUGE METERIAL (MOHR-COULOMB MODEL)
  (ENTER 3 OR 4)
? 3                                                                    USER
   INPUT MATERIAL PROPERTIES:
    (COHESION AND FRICTION ANGLE CAN BE ASSIGNED 0.0)
    IF THEY ARE UNNECESSARY.)
    YOUNGS MODULUS =
? 200000                                                               USER
    SHEAR MODULUS =
? 76923                                                                USER
    COHESION OF SOLID MATERIAL =
? 0                                                                    USER
    INTERNAL FRICTION ANGLE OF SOLID MATERIAL =
? 0                                                                    USER
    COHESION OF BROKEN MATERIAL =
? 0                                                                    USER
    INTERNAL FRICTION ANGLE OF BROKEN MATERIAL =
? 0                                                                    USER
    IS THIS A GOB/INSERTED MATERIAL TYPE (Y/N)
? N                                                                    USER
  DATA ENTERED FOR MATERIAL NUMBER: 1
   YOUNGS MODULUS: 200000
   SHEAR MODULUS: 76923
   COHESION     : 0.
   INTERNAL ANGLE: 0.
   COHESION BROKEN: 0.
   INTERNAL ANG BROKEN: 0.
   GOB MATERIAL: N
   IS THIS OK(Y/N)
? Y                                                                    USER
CARD TYPE 4
MATERIAL NUMBER: ---2
MATERIAL CODE:    ---B
IS THIS A
 GOUGE MATERIAL(MOHR-COULOMB MODEL)?
3 = "NORMAL MATERIALS"
    <SEAM MATERIALS,GOB, OR "INSERTED" MATERIALS)
```

```
 4 = GOUGE MATERIAL (MOHR-COULOMB MODEL)
 (ENTER 3 OR 4)
? 3                                                               USER
  INPUT MATERIAL PROPERTIES:
   (COHESION AND FRICTION ANGEL CAN BE ASSIGNED 0.0
   IF THEY ARE UNNECESSARY.)
   YOUNG'S MODULUS =
? 5000                                                             USER
   SHEAR MODULUS =
? 1786                                                             USER
   COHESION OF SOLID MATERIAL =
? 0                                                                USER
   INTERNAL FRICTION ANGLE OF SOLID MATERIAL =
? 0                                                                USER
   COHESION OF BROKEN MATERIAL =
? 0                                                                USER
   INTERNAL FRICTION ANGLE OF BROKEN MATERIAL =
? 0                                                                USER
  IS THIS A GOB/INSERTED MATERIAL TYPE(Y/N)
? Y                                                                USER
  DATA ENTERED FOR MATERIAL NUMBER: 2
   YOUNG'S MODULUS: 5000.
   SHEAR MODULUS: 1786
   COHESION     : 0.
   INTERNAL ANGLE: 0.
   COHESION BROKEN: 0.
   INTERNAL ANG BROKEN: 0.
  IS THIS OK(Y/N)
? Y                                                                USER
 CARD TYPE 5---
INPUT THE PRIMITIVE STRESS DATA
  THE STRESS TENSOR THAT DESCRIBES A STRESS STATE
  THAT EXISTS PRIOR TO MINING IS DEFINED AS LINEAR
  FUNCTIONS OF GLOBAL X3 DIRECTION. WITH COMPRESSION
  BEING POSITIVE, THE EQUATIONS ARE OF THE FORM:
               STRESS(1,1) = A11-B11*X3
               STRESS(1,2) = A12-B12*X3   ETC.
               (X3 IS NEGATIVE)
  AFTER CONVERSION(IF ANY)
    AXX MUST BE >= -.9999 AND <= 99999.
    BXX MUST BE >= -.9999 AND <= 9.9999
 INPUT STRESS DATA IN PSI ASSUMING X3 IS IN FEET;
 THE MESH GENERATOR WILL DIVIDE YOUR BXX VALUES BY 12
 TO MAINTAIN CONSISTENT UNITS SINCE COORDINATES
 HAVE BEEN CONVERTED TO INCHES
  INPUT A11:
? 0                                                                USER
  INPUT B11:
? 0.55                                                             USER
  INPUT A12:
? 0                                                                USER
```

```
  INPUT B12:
? 0                                                         USER
  INPUT A13:
? 0                                                         USER
  INPUT B13:
? 0                                                         USER
  INPUT A22:
? 0                                                         USER
  INPUT B22:
? 0.55                                                      USER
  INPUT A23:
? 0                                                         USER
  INPUT B23:
? 0                                                         USER
  INPUT A33:
? 0                                                         USER
  INPUT B33:
? 1.1                                                       USER
  PRIMITIVE STRESS DATA ENTERED
 A11,B11,.......B33
 0. 0.55 0. 0. 0. 0. 0. 0.55 0. 0. 0. 1.1
  IS THIS OK(Y/N)
? Y                                                         USER
 CARD TYPE 6---
   MODEL DATA
 IF SYMMETRY IS SPECIFIED. THE AXES OF SYMMETRY
 PASS THROUGH THE CENTER OF THE MODEL
 WHAT IS THE SYMMETRY CODE
   1 - NO SYMMETRY
   2 - COLUMN SYMMETRY(VERTICAL AXIS OF SYMMETRY)
   3 - ROW SYMMETRY(HORIZONTAL AXIS OF SYMMETRY)
   4 - COLUMN AND ROW SYMMETRY
 ENTER 1,2,3, OR 4
? 1                                                         USER
   WHAT IS THE BLOCK WIDTH
  (INPUT WIDTH IN FEET: YOUR DATA WILL BE CONVERTED
  TO INCHES.)
? 100                                                       USER
   HOW MANY BLOCKS LIE ALONG THE LOCAL X AXIS IN
   EACH SEAM(TOTAL GRID-COARSE + FINE)
   IF SYMMETRY WAS SPECIFIED. INPUT ONLY THE UNIQUE
   PORTION (QUADRANT OR HALF) OF THE MESH.
? 8                                                         USER
   HOW MANY BLOCKS LIE ALONG THE LOCAL Y-AXIS
   IN EACH SEAM(TOTAL GRID-COARSE + FINE)
   IF SYMMETRY WAS SPECIFIED. INPUT ONLY THE UNIQUE
   PORTION (QUADRANT OR HALF) OF THE MESH.
? 8                                                         USER
   FINE MESH STARTING BLOCK X AXIS
```

```
? 4                                                                        USER
   FINE MESH ENDING BLOCK X AXIS
? 5                                                                        USER
   FINE MESH STARTING BLOCK Y AXIS
? 2                                                                        USER
   FINE MESH ENDING BLOCK Y AXIS
? 7                                                                        USER
 AT THIS POINT THE MODEL WILL BE PLOTTED OUT ON THE TEKTRONIX
 SCREEN.  THE USER WILL SEE A PROMPT AT THE TOP LEFT HAND CORNER
 OF THE SCREEN.
 IF THE USERS RESPONSE TO THE QUESTION "IS THE MODEL OK" IS NO
 THEN THE PROMPTS FOR CARD TYPE 6 ARE REPEATED AND THEN THE
 MODEL IS PRESENTED ON THE SCREEN AGAIN.
  IS MODEL OK(Y/N)
 ? Y                                                                       USER
  DO YOU WANT A HARD COPY(Y/N)
 ? N                                                                       USER
 HOW MANY OFF-SEAM PLANES WILL BE USED TO
 OBTAIN ADDITIONAL STRESS AND DISPLACEMENT DATA
? 1                                                                        USER
  MODEL DATA ENTERED IS:
  BLOCK WIDTH: 100.
  X-AXIS BLOCKS: 8.
  Y-AXIS BLOCKS: 8.
  FINE MESH X   : 4
                : 5
  FINE MESH Y   : 2
                : 7
  SYMMETRY CODE: 1
  NUM OFFSEAM   : 1
  IS THIS OK(Y/N)
? Y                                                                        USER
 CARD TYPE 7---
 EXTRACTION CODE "0" IS ZERO EXTRACTION
 (THIS CODE IS IRRELEVANT FOR GOB OR OPENINGS)
 ENTER NUMBER OF ADDITIONAL
 EXTRACTION RATIO CODES(0 TO 9)
? 1                                                                        USER
   ENTER EXTRACTION RATIO FOR CODE 1
      (FRACTIONAL NUMBER > 0 AND < 1)
? .2                                                                       USER
  EXTRACTION DATA ENTERED IS:
  NUMBER OF NON-ZERO EXTRACTION CODES: 1
  EXTRACTION RATIOS:.2
  IS THIS OK(Y/N)
? Y                                                                        USER
  CARD TYPE 8---
 SEAM NUMBER: 1
 WHAT ARE THE GLOBAL COORDINATES OF THE GRID
```

```
 ORIGIN
 (INPUT IN FEET;   YOUR DATA WILL BE CONVERTED
 TO INCHES.
  XO =
? 0                                                              USER
  YO =
? 0                                                              USER
  ZO =
? -1080                                                          USER
  THICKNESS =
? 10                                                             USER
  GLOBAL COORDINATES FOR SEAM 1
 XO: 0.
 YO: 0.
 ZO: -1080.
 THICKNESS: 10.
 IS THIS OK(Y/N)
? Y                                                              USER
  CARD TYPE 8---
 SEAM NUMBER: 2
 WHAT ARE THE GLOBAL COORDINATES OF THE GRID
 ORIGIN
 (INPUT IN FEET;   YOUR DATA WILL BE CONVERTED
 TO INCHES.
  XO =
? 0                                                              USER
  YO =
? 0                                                              USER
  ZO =
?  1000                                                          USER
   MUST BE A NEGATIVE NUMBER
  ZO =
? -1000
  THICKNESS =
?  8                                                             USER
  GLOBAL COORDINATES FOR SEAM 2
 XO: 0.
 YO: 0.
 ZO: -1000.
 THICKNESS: 8.
 IS THIS OK(Y/N)
? Y                                                              USER
  CARD TYPE 9---
  GRID ORIENTATION
 SPECIFY STRIKE, DIP AND OFFSET ANGLE
 STRIKE ANGLE IS IN DEGREES COUNTERCLOCKWISE FROM
 GLOBAL Y
 (COUNTERCLOCKWISE IS MEASURED LOOKING DOWN FROM THE
 SURFACE)
```

```
 WHAT IS THE STRIKE ANGLE
? 0                                                                USER
 POSITIVE DIP IS TOWARD THE ROTATED X DIRECTION
 WHAT IS THE ANGLE OF DIP
? 0                                                                USER
 IN THE PLANE OF THE SEAM, THE OFFSET ANGLE
 IS MEASURED COUNTERCLOCKWISE FROM THE DIP DIRECTION.
 TO THE ENTRY LINE ( THE LOCAL X-AXIS OF YOUR MESH
 WHAT IS THE OFFSET ANGLE?
? 0                                                                USER
 ANGLES ENTERED ARE:
  STRIKE: 0.
     DIP: 0.
  OFFSET: 0.
  IS THIS OK(Y/N)
? Y                                                                USER
  CARD TYPE 10---
    PROGRAM FLOW PARAMETERS
    THE OVER-RELAXATION FACTOR CAN BE ANY NUMBER
    FROM 1.0 TO 1.99 (1.35 IS OFTEN BEST).
    OVER-RELAXATION FACTOR =
? 1.35                                                             USER
    WHAT RUN NUMBER IS THIS
? 1                                                                USER
   HOW MANY ITERATIONS WERE COMPLETED IN PREVIOUS RUNS
?  0                                                               USER
   MAXIMUM NUMBER OF NEW ITERATIONS FOR THIS RUN =
?  100                                                             USER
   IS A PREVIOUS COEFFICIENT MAXRIX TO BE USED
   ENTER (1 = YES, 0 = NO)
? 0                                                                USER
 FLOW PARAMETERS ENTERED ARE:
 RELAXATION FACTOR: 1.35
 RUN NUMBER: 1
 PREV ITERATIONS: 0.
 MAX NEW ITERATIONS: 100.
 PREV COFFICIENTS:   0.
   IS THIS OK(Y/N)
? Y                                                                USER
  CARD TYPE 15---
  OFFSEAM GRID DETAILS:  PLANE NO. 1
  INPUT COORDINATES AND WIDTH IN FEET:
  AND THEY WILL BE CONVERTED TO INCHES
      LOCAL ORIGIN
            XO =
? 300                                                              USER
            YO =
? 400                                                              USER
            ZO =
```

```
? -1089                                                                  USER
   BLOCK WIDTH ALONG THE LOCAL X AXIS=
? 20                                                                     USER
   BLOCK WIDTH ALONG THE LOCAL Y AXIS=
? 20                                                                     USER
   NUMBER OF BLOCKS IN LOCAL X DIRECTION=
? 10                                                                     USER
   NUMBER OF BLOCKS IN LOCAL Y DIRECTION=
? 1                                                                      USER
  OFFSEAM GRID DETAILS FOR PLANE NO: 1
  X0 = 300.
  Y0 = 400.
  Z0 = -1089.
  X-AXIS BLOCK WIDTH: 20.
  Y-AXIS BLOCK WIDTH: 20.
  X-AXIS NUM OF BLOCKS: 10.
  Y-AXIS NUM OF BLOCKS: 1.
  IS THIS OK(Y/N)
 ? Y                                                                     USER
  CARD TYPE 15A---
  INPUT PLANE ORIENTATION
 SPECIFY STRIKE, DIP AND OFFSET ANGLE
 STRIKE ANGLE IS IN DEGREES COUNTERCLOCKWISE FROM
 GLOBAL Y
 (COUNTERCLOCKWISE IS MEASURED LOOKING DOWN FROM THE
 SURFACE
 WHAT IS THE STRIKE ANGLE
? 0                                                                      USER
 POSITIVE DIP IS TOWARD THE ROTATED X DIRECTION
 WHAT IS THE ANGLE OF DIP
? 0                                                                      USER
 IN THE PLANE OF THE SEAM, THE OFFSET ANGLE
 IS MEASURED COUNTERCLOCKWISE FROM THE DIP DIRECTION.
 TO THE ENTRY LINE ( THE LOCAL X-AXIS OF YOUR MESH
 WHAT IS THE OFFSET ANGLE?
? 0                                                                      USER
 ANGLES ENTERED ARE:
  STRIKE: 0.
     DIP: 0.
  OFFSET: 0.
  IS THIS OK(Y/N)
? Y                                                                      USER
THE MESH GENERATOR WILL NOW DISPLAY THE FINE MESH AREA ON THE
SCREEN AND ALLOW THE USER TO INSERT IN EACH ELEMENT THE APPROPRIATE
MATERIAL PROPERTY CODE.  THE USER IS FIRST PROMPTED FOR THE DEFAULT
MATERIAL PROPERTY CODE WHICH WILL FILL THE FINE MESH AREA.
THE FINE MESH AREA WILL BE MODELED FIRST
FOLLOWED BY THE COARSE MESH AREA
ENTER THE DEFAULT PROPERTY FOR SEAM    1
```

```
   ? A                                                               USER
    1 = OPENING
    2 = RIGID;
    A-Z = MATERIALS;
     ENTER LETTER
     OR NUMBER
     TO SUBSTITUTE
? 1                                                                  USER
     MOVE CURSOR TO
     THE TWO OPPOSITE
     CORNERS OF THE
     ZONE TO CHANGE
     ENTER "X"<RETURN>
     AT EACH CORNER
     ENTER "E"<RETURN>
     WHEN DONE
     IF USING A 4054
     TERMINAL, DO NOT
     PRESS <RETURN> AFTER
     ENTERING "X" OR "E"
     THE USER WILL NOW CHANGE THE MODEL BY USING THE GRAPHIC INPUT
     CURSOR AS PER THE ABOVE INSTRUCTIONS.  WHEN THE USER ENTERS AN
     "E" INDICATING THAT THE CHANGES ARE COMPLETED, THE PROGRAM WILL
     PROMPT AS FOLLOWS:
     CHANGES(Y/N)
 ? N                                                                 USER
     DO YOU WANT
     A HARD COPY(Y/N)
 ? N                                                                 USER
       THE SAME PROCEDURE IS FOLLOWED IN MODELING THE COARSE
       MESH AREA AND THEN IN MODELING THE EXTRACTION RATIOS FOR
       THE COARSE MESH BLOCKS.
       WHEN THE ENTIRE SEAM HAS BEEN MODELED, THE PROGRAM WILL
       PROMPT THE USER AGAIN.
       IF SEAM 1
       IS NOT O.K.
       YOU MAY REMODEL
       THE ENTIRE MESH
       FOR THIS SEAM.
       IS SEAM 1 O.K.(Y/N)
   ? Y                                                               USER
       SEAM 2 WILL BE MODELED IN THE SAME MANNER AS SEAM 1.
       THE USER WILL BE PROMPTED AS TO WHETHER SEAM 2 IS
       ALLRIGHT.  IF NOT THEN SEAM 2 WILL BE COMPLETELY
       REMODELED AGAIN.
      ENTER FILE NAME (FOR FILE TO BE CREATED)
   ? SAMPLE                                                          USER
   ...NOW WRITING CONTROL CARDS
   ...NOW WRITING MATERIAL CODES FOR SEAM 1
```

```
...NOW WRITING MATERIAL CODES FOR SEAM 2
...NOW WRITING STATUS CODES FOR SEAM 1
...NOW WRITING STATUS CODES FOR SEAM 2
...NOW WRITING COARSE MESH MATERIAL CODES FOR SEAM 1
...NOW WRITING COARSE MESH MATERIAL CODES FOR SEAM 2
...NOW WRITING COARSE MESH EXTRACTION CODES FOR SEAM 1
...NOW WRITING COARSE MESH EXTRACTION CODES FOR SEAM 2
...NOW WRITING OFF-SEAM CONTROL CARDS
    WHAT DO YOU WANT TO DO NOW
          TYPE 0---TO END THIS PROGRAM NOW
          TYPE 1---TO GENERATE A DATA FILE
? 0                                                        USER
 READY

 ---------------------------------------------
 -      END OF SAMPLE TERMINAL SESSION       -
 ---------------------------------------------
```

```
 ----------------------------------------
 -  LISTING OF PROCEDURE FILE PRGENM    -
 ----------------------------------------
```

```
.PROC,PRGENM*I.
CLEAR(*,PRGENM)
RFL,77000.
GET,BGNMLBM/UN=BM0001L.
ATTACH,PLOT10/UN=GRAFLIB.
LDSET,LIB=PLOT10.
BGNMLBM.
REVERT,NOLIST.
```

APPENDIX C.--MULSIM/BM STRUCTURE DIAGRAM AND PROGRAM LISTING

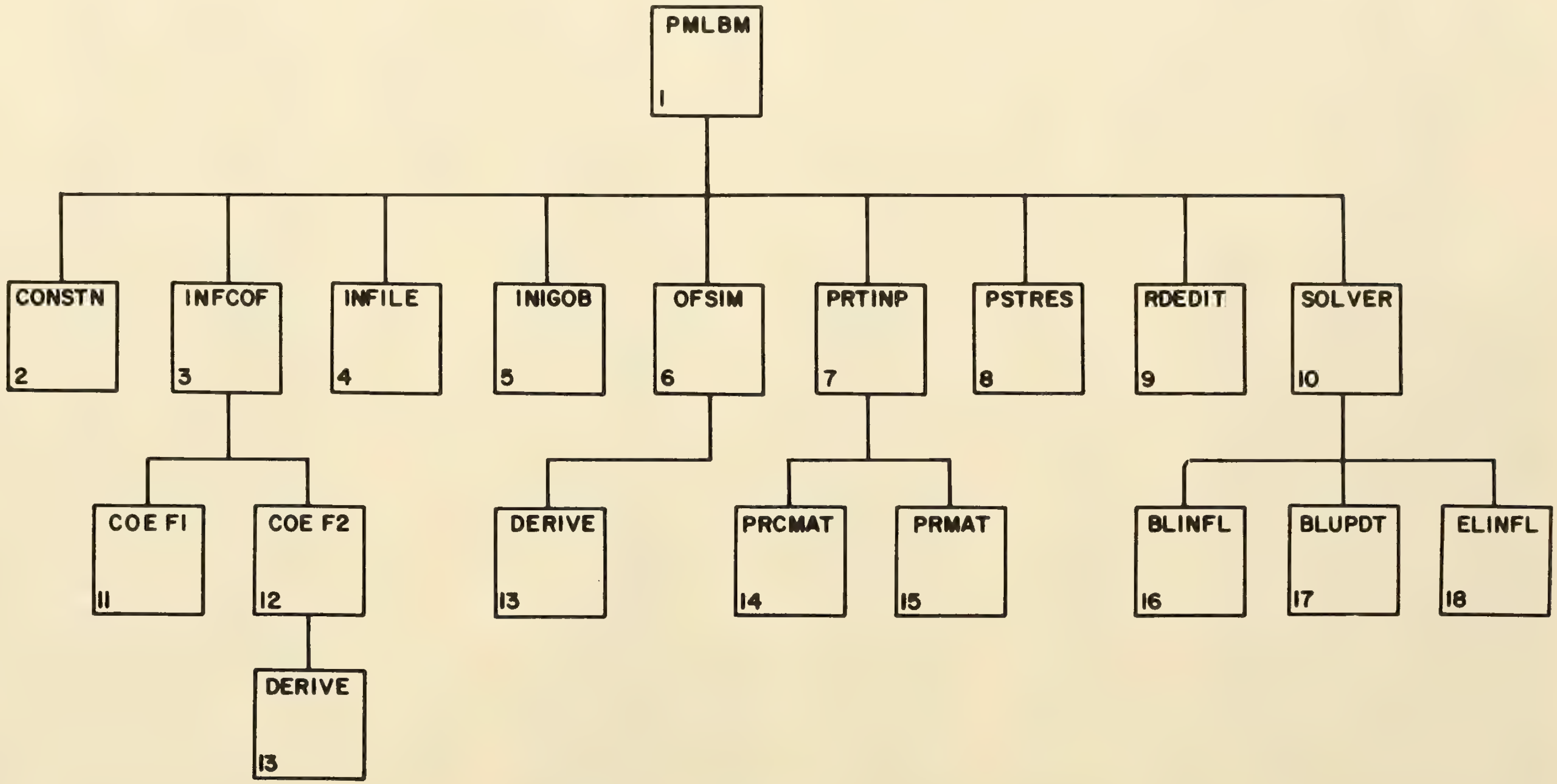


FIGURE C-1.—MULSIM/BM structure diagram.

```
      PROGRAM PMLBM (INPUT, OUTPUT, COFFS, TAPE16=COFFS, TAPE13,
     # TAPE30, COFFSD, TAPE20=COFFSD, DISC, TAPE10=DISC, TAPE12,
     #                 SEAM1, TAPE31=SEAM1, SEAM2, TAPE32=SEAM2,
     #                CSEAM1, TAPE41=CSEAM1, CSEAM2, TAPE42=CSEAM2)
C
C   BUREAU OF MINES FINAL 2-SEAM VERSION OF MULSIM/BM -- NOV 5,1985
C
C   THIS VERSION ALLOWS A COARSE MESH REGION TO BE DEFINED AROUND
C   A FINE MESH REGION.  THE COARSE MESH REGION CONSISTS OF UNDIVIDED
C   BLOCKS.  EACH BLOCK MAY BE ASSIGNED A MATERIAL PROPERTY CODE AS
C   WELL AS AN EXTRACTION RATIO CODE.  THE FINE MESH REGION CONSISTS
C   OF BLOCKS SUBDIVIDED INTO 5 ROWS OF 5 ELEMENTS PER ROW.  EACH FINE
C   MESH ELEMENT IN EACH BLOCK MAY BE ASSIGNED A MATERIAL PROPERTY
C   CODE.
C
C
      COMMON/BK1/    DUB(3200), DVB(3200), DWB(3200),DWBI(3200),
     #               DE(2100),  INDEX(805),  KODE(25), INDX13(805)
     #               ,NGOB(3200)
      COMMON/BK2/    PO1(4), PO2(4), PO3(4),
     #               P1XI,   P2XI,   P3XI,
     #               P1ET,   P2ET,   P3ET
      COMMON/BK3/    EN(3,3), IMAX, JMAX
      COMMON/BK4/    S13U(3300), S23U(3300), S33U(3300),
     #               S23V(3300), S33V(3300), S33W(3300),
     #               CF(3,27,4)
      COMMON/BK5/    D1U(3300), D2U(3300), D3U(3300),
     #               D2V(3300), D3V(3300),
     #               D1W(3300), D2W(3300), D3W(3300)
      COMMON/CONSTS/ PI, FACT, V1, V2, CONS, COND, CON, IAMAT(28),
     #               HEW, HBW, EW, RLIM, NBXE, NBT, KB, NTR, ITMINI,
     #               ITMAXI, IPHASE, JOG, ID(4,4), LQR, KI, KE, NCUT,
     #               ACC, NONE, KD0, KD1, KD2, KD3, KD4, KD5, KD6
     #               , NELMX, NELMY, NBLKX, NBLKY, NBLKFM
```

```
      COMMON/IN1/   TITLE(20), V, E, NSEAM, NMATS
      COMMON/IN2/   ES(28), GS(28), COHES(28),
     #              PHI(28), COHES1(28), PHI1(28), IGOB(28)
      COMMON/IN3/   A11, A12, A13, A22, A23, A33,
     #              B11, B12, B13, B22, B23, B33
      COMMON/IN4/   BW, NBXI, NBET, NSYM, NOSP, IFXS, IFXE, IFYS, IFYE
      COMMON/IN4A/  IEX, EXR(10)
      COMMON/IN5/   XO(4), YO(4), ZO(4), THIKNS(4)
      COMMON/IN6/   STRANG, DIPANG, OFSANG
      COMMON/IN7/   ORF, ITMAX, NRUN, ITP, NGRID
      COMMON/IN9/   MKOD(2403), MCKOD(3200), MCEKOD(3200)
      COMMON/AK1/   S13PB, S23PB, S33PB, UPB, VPB, WPB,
     #              S13PE, S23PE, S33PE, UPE, VPE, WPE
      COMMON/AK2/   SIG1(25), SIG2(25), SIG3(25),
     #              UPOS(25), VPOS(25), WPOS(25),
     #              UNEG(25), VNEG(25), WNEG(25)
C
      DATA DUB, DVB, DWB, DE, MKOD, KODE/11700*0.0,2428*0/
C
      DATA DWBI/3200*0.0/
C   INITIALIZE ARRAYS TO BE WRITTEN ON TAPE16
C
      DATA S13U,S23U,S33U,S23V,S33V,S33W,CF/20124*0.0/
```

PROGRAM PMLBM 74/855 OPT=2 FTN 4.8+587 86/02/05. 13.56.15 PAGE 2

```
C
C   INITIALIZE ARRAYS TO BE WRITTEN ON TAPE20
C
      DATA D1U,D2U,D3U,D2V,D3V,D1W,D2W,D3W/26400*0.0/
C
C
C..REWIND ALL SEQUENTIAL FILES USED BY PROGRAM
C
      REWIND 12
      REWIND 16
```

```
      REWIND 20
      REWIND 31
      REWIND 32
      REWIND 41
      REWIND 42
      CALL SECOND(TIMB)
C
************************************************************************
*                  READ AND EDIT INPUT PARAMETERS                      *
************************************************************************
C
      CALL RDEDIT(MANY)
C
C
**********************************************************************
*                   PRINT INPUT PARAMETERS
**********************************************************************
C
      CALL PRTINP
C
C  OUTPUT THE CPU TIME AT THIS STAGE
C
      CALL SECOND(TIME)
      TIM=TIME-TIMB
      PRINT 10000,TIM
      TIMB=TIME
C
************************************************************************
*              DEFINE CONSTANTS AND INITIAL PARAMETERS                 *
************************************************************************
C
      CALL CONSTN
C
*************************************************************************
C
```

```
      *         DEFINE THE MASS STORAGE FILE FOR RIDE AND CLOSURES            *
      *                          AND INITIALIZE                               *
      *************************************************************************
      C
            CALL INFILE
      C
      C
      ************************************************************************
      *              OBTAIN PRIMITIVE STRESSES AND THEIR GRADIENTS           *
      ************************************************************************
      C
            CALL PSTRES
PROGRAM PMLBM 74/855 OPT=2 FTN 4.8+587  86/02/05. 13.56.15  PAGE 3

      C
      C  OUTPUT THE CPU TIME AT THIS STAGE
      C
            CALL SECOND(TIME)
            TIM=TIME-TIMB
            TIMB=TIME
            PRINT 10010,TIM
      C
      ************************************************************************
      *                      OBTAIN THE INFLUENCE COEFFICIENTS               *
      *                  AND THE RIDE AND CLOSURE CORRECTION FACTORS         *
      ************************************************************************
      C
      C     INITIALIZE THE COMMON AREA USED TO HOLD
      C     MATERIAL PROPERTIES TEMPORARILY
      C
            DO 100 I=1,3300
                  S13U(I) = 0.0
                  S23U(I) = 0.0
                  S33U(I) = 0.0
        100 CONTINUE
```

```
C
      CALL INFCOF
C
C
C  OUTPUT CPU TIME AT THIS STAGE
C
      CALL SECOND(TIME)
      TIM=TIME-TIMB
      TIMB=TIME
      PRINT 10020,TIM
C
C..INITIALIZE CLOSURES FOR GRID BLOCKS
C
C
      CALL INIGOB
C
C   STOP IF NO. OF ITERATIONS <=0  (DATA CHECK ONLY)
C
      IF (ITMAX.GT.0) GOTO 420
      PRINT 27000
27000 FORMAT(//* DATA CHECK OPTION SELECTED:  NO. ITERATIONS IS ZERO.*//
     A* (PROGRAM STOPS)*)
      STOP
  420 CONTINUE
C
C
************************************************************************
*                  SOLVE FOR UNKNOWN CLOSURES AND RIDES                *
************************************************************************
C
      CALL SOLVER(IND1,IND2,IND3,MANY,JTER,IND,ERROR
     #           ,INDB1,INDB2,INDB3,ERRORB,INDB)
C
C  WRITE THE UPDATED BLOCK AVERAGE RIDE AND CLOSURE AND STATUS KODE
C
```

```
      C  VALUES ON FILE TAPE10=DISC.
PROGRAM PMLBM 74/855 OPT=2 FTN 4.8+587   86/02/05. 13.56.15  PAGE 4

      C
            CALL WRITMS(10,DUB(1),NBT,1,1)
            CALL WRITMS(10,DVB(1),NBT,2,1)
            CALL WRITMS(10,DWB(1),NBT,3,1)
            CALL WRITMS(10,MKOD,KB,4,1)
      C
      C  OUTPUT THE CPU TIME AT THIS STAGE
      C
            CALL SECOND(TIME)
            TIM=TIME-TIMB
            TIMB=TIME
            PRINT 10030,TIM
      C
      ************************************************************************
      *       COMPUTE THE DISPLACEMENTS AND STRESSES AT ALL THE SEAM         *
      *       AND SPECIFIED OFF-SEAM LOCATIONS AND OUTPUT THE RESULTS        *
      ************************************************************************
      C
      C  FIRST AT THE SEAM LOCATIONS--
      C  REASSIGN THE PROGRAM ROUTES AND RESET SOME OF THE CONSTANTS
      C
            ITMINI=ITMAXI
            IPHASE = 6HOUTPUT
            KD0=0
            KD6=1
            IND1=IND1*MANY
            IND2=IND2*MANY
            IND3=IND3*MANY
      C
      C  OUTPUT THE ITERATION PROCESS STATISTICS
      C
            PRINT 10040,JTER
```

```
          IF(IND.GT.0) PRINT 10050,IND1,IND2,IND3
          PRINT 10060,ERROR
          PRINT 10070
          IF(INDB.GT.0) PRINT 10055,INDB1,INDB2,INDB3
          PRINT 10065,ERRORB
    C
    C..ITERATE FOR TOTAL CLOSURES AND RIDES
    C
          CALL SOLVER(IND1,IND2,IND3,MANY,JTER,IND,ERROR
         #            ,INDB1,INDB2,INDB3,ERRORB,INDB)
    C
    C  OUTPUT THE CPU TIME AT THIS STAGE
    C
          CALL SECOND(TIME)
          TIM=TIME-TIMB
          TIMB=TIME
          PRINT 10080,TIM
    C
    C..OUTPUT RESULTS FOR COARSE MESH BLOCKS
    C
          REWIND 30
    C
          DO 500 NP=1,NSEAM
            LPBO = (NP-1) * NBXE
            PRINT 10200, NP
```



```
            DO 400 JP=1,JMAX
              DO 300 IP = 1,IMAX
                LPB = LPBO + (JP-1) * NBET + IP
                MATYP  = MCKOD(LPB)
                MATYP  = MATYP + 2
                IF(MATYP .EQ. 29) MATYP = 1
                IF(MATYP .EQ. 30) MATYP = 2
                ICODE = 1H*
```

```
              IF( IP .GE. IFYS .AND. IP .LE. IFYE .AND.
     #            JP .GE. IFXS .AND. JP .LE. IFXE ) GO TO 200
              ICODE = IAMAT(MATYP)
  200         READ (30) UPOSBL, VPOSBL, WPOSBL,
     #                  UNEGBL, VNEGBL, WNEGBL,
     #                  SIG1BL, SIG2BL, SIG3BL
              PRINT 10300, JP, IP, ICODE, DUB(LPB), DVB(LPB),
     #                     DWB(LPB), UPOSBL, VPOSBL, WPOSBL, UNEGBL,
     #                     VNEGBL, WNEGBL, SIG1BL, SIG2BL, SIG3BL
C
C.. WRITE OUT COARSE SEAM RESULTS TO COARSE MESH FILE
C
              IUNIT = NP + 40
              WRITE(IUNIT,10300) JP, IP, ICODE, DUB(LPB), DVB(LPB),
     #                     DWB(LPB), UPOSBL, VPOSBL, WPOSBL, UNEGBL,
     #                     VNEGBL, WNEGBL, SIG1BL, SIG2BL, SIG3BL
  300           CONTINUE
  400         CONTINUE
  500 CONTINUE

C
C  THEN AT THE OFF-SEAM LOCATIONS
C
         IF(NOSP .EQ. 0) GO TO 999
           PRINT 10090,NOSP
           CALL OFSIM(NOSP)
C
C..OUTPUT THE CPU TIME AT THIS STAGE
C
           CALL SECOND(TIME)
           TIM=TIME-TIMB
           TIMB=TIME
           PRINT 10100,TIM
C
C
```

```
  999 IF(ITP.EQ.0) CALL CLOSMS(10)
C
C
C..OUTPUT FORMATS
C
10000 FORMAT(/*  TIME TAKEN TO READ AND PRINT INPUT VARIABLES
     1              --*,F10.4,* SECOND*)
10010 FORMAT(/*  TIME TAKEN TO DEFINE THE CONSTANTS, INITIALIZE THE CLOS
     1URE AND */*  RIDE VALUES AND OBTAIN THE PRIMITIVE STRESSES
     2                --*,F10.4,* SECOND*)
10020 FORMAT(/*  TIME TAKEN TO COMPUTE/RETRIEVE THE INFLUENCE COEFFICIEN
     1TS            --*,F10.4,* SECOND*)
10030 FORMAT(/*  TIME TAKEN TO SOLVE FOR UNKNOWN CLOSURES AND RIDES
     1              --*,F10.4,* SECOND*)
```

PROGRAM PMLBM 74/855 OPT=2 FTN 4.8+587 86/02/05. 13.56.15 PAGE 6

```
10040 FORMAT(/*  NO. OF ITERATIONS COMPLETED IN ALL THE RUNS   =*,I8)
10050 FORMAT (/*  INSUFFICIENT ACCURACY--*,I4,*  ELEMENTS AFFECTED IN RI
     1DE ALONG XI AXIS*/*                          *,I4,*  ELEMENTS AFFEC
     2TED IN RIDE ALONG ET AXIS*/*                           *,I4,*  ELEME
     3NTS AFFECTED IN CLOSURE*)
10055 FORMAT (/*  INSUFFICIENT ACCURACY--*,I4,*  BLOCKS AFFECTED IN RIDE
     1 ALONG XI AXIS*/*                          *,I4,*  BLOCKS AFFECTED
     2IN RIDE ALONG ET AXIS*/*                          *,I4,*  BLOCKS AF
     3FECTED IN CLOSURE*)
10060 FORMAT(/*  MAXIMUM ERROR IN THE RIDE AND CLOSURE VALUES =*,F10.7)
10065 FORMAT(/*  MAXIMUM ERROR IN THE RIDE AND CLOSURE BLOCK VALUES =*
     #     ,F10.7)
10070 FORMAT(/*  CLOSURE, RIDE AND DISPLACEMENT VALUES ARE IN THE UNITS
     1 OF INPUT LINEAR DIMENSIONS*/*  (VIZ,XO,YO,ZO,HW...ETC) AND STRESS
     2ES ARE IN THE UNITS OF MODULUS OF ELASTICITY (E).* )
10080 FORMAT(/*  TIME TAKEN TO OUTPUT THE RESULTS FOR SEAM LOCATIONS
     1              --*,F10.4,* SECOND*)
10090 FORMAT(///*  DISPLACEMENTS AND STRESSES AT OFF-SEAM ELEMENTS.*//
     1*  TOTAL NO. OF OFF-SEAM PLANES      --*,I4)
```

```
      10100 FORMAT(/*  TIME TAKEN TO OUTPUT THE OFF-SEAM DISPLACEMENTS AND STR
           1ESSES       --*,F10.4,* SECOND*)
      10200 FORMAT(/*1 NO.OF SEAM--*,I4,//* BLOCK   BLOCK  *
           1                              //*   COL     ROW  CODE   RIDE1   RIDE
           12 CLOSURE   U1POS   U2POS   U3POS   U1NEG   U2NEG   U3NEG     SIGS
           21     SIGS2       SIGN*/)
      10300 FORMAT(2I8,1X,A1,2X,9F8.3,3F10.2)
      C
            STOP
            END
```

```
 SYMBOLIC REFERENCE MAP (R=1)

ENTRY POINTS
31051   PMLBM

VARIABLES       TYPE              RELOCATION
   103  ACC     REAL                  CONSTS          0  A11     REAL                IN3
     1  A12     REAL                  IN3             2  A13     REAL                IN3
     3  A22     REAL                  IN3             4  A23     REAL                IN3
     5  A33     REAL                  IN3             0  BW      REAL                IN4
     6  B11     REAL                  IN3             7  B12     REAL                IN3
    10  B13     REAL                  IN3            11  B22     REAL                IN3
    12  B23     REAL                  IN3            13  B33     REAL                IN3
 46530  CF      REAL      ARRAY       BK4            70  COHES   REAL      ARRAY     IN2
   160  COHES1  REAL      ARRAY       IN2             6  CON     REAL                CONSTS
     5  COND    REAL                  CONSTS          4  CONS    REAL                CONSTS
 31000  DE      REAL      ARRAY       BK1             1  DIPANG  REAL                IN6
     0  DUB     REAL      ARRAY       BK1          6200  DVB     REAL      ARRAY     BK1
 14400  DWB     REAL      ARRAY       BK1         22600  DWBI    REAL      ARRAY     BK1
     0  D1U     REAL      ARRAY       BK5         40164  D1W     REAL      ARRAY     BK5
  6344  D2U     REAL      ARRAY       BK5         23254  D2V     REAL      ARRAY     BK5
 46530  D2W     REAL      ARRAY       BK5         14710  D3U     REAL      ARRAY     BK5
 31620  D3V     REAL      ARRAY       BK5         55074  D3W     REAL      ARRAY     BK5
```

```
VARIABLES          TYPE              RELOCATION
   25   E          REAL                 IN1         0   EN        REAL      ARRAY    BK3
32145   ERROR      REAL                         32151   ERRORB    REAL
    0   ES         REAL      ARRAY      IN2        45   EW        REAL               CONSTS
    1   EXR        REAL      ARRAY      IN4A        1   FACT      REAL               CONSTS
   34   GS         REAL      ARRAY      IN2        44   HBW       REAL               CONSTS
   43   HEW        REAL                 CONSTS  32137   I         INTEGER
    7   IAMAT      INTEGER   ARRAY      CONSTS  32161   ICODE     INTEGER
   57   ID         INTEGER   ARRAY      CONSTS      0   IEX       INTEGER            IN4A
    6   IFXE       INTEGER              IN4         5   IFXS      INTEGER            IN4
   10   IFYE       INTEGER              IN4         7   IFYS      INTEGER            IN4
  250   IGOB       INTEGER   ARRAY      IN2        11   IMAX      INTEGER            BK3
32144   IND        INTEGER                      32152   INDB      INTEGER
32146   INDB1      INTEGER                      32147   INDB2     INTEGER
32150   INDB3      INTEGER                      35064   INDEX     INTEGER   ARRAY    BK1
36562   INDX13     INTEGER   ARRAY      BK1     32140   IND1      INTEGER
32141   IND2       INTEGER                      32142   IND3      INTEGER
32156   IP         INTEGER                         55   IPHASE    INTEGER            CONSTS
    1   ITMAX      INTEGER              IN7        54   ITMAXI    INTEGER            CONSTS
   53   ITMINI     INTEGER              CONSTS      3   ITP       INTEGER            IN7
32173   IUNIT      INTEGER                         12   JMAX      INTEGER            BK3
   56   JOG        INTEGER              CONSTS  32155   JP        INTEGER
32143   JTER       INTEGER                         51   KB        INTEGER            CONSTS
  105   KD0        INTEGER              CONSTS    106   KD1       INTEGER            CONSTS
  107   KD2        INTEGER              CONSTS    110   KD3       INTEGER            CONSTS
  111   KD4        INTEGER              CONSTS    112   KD5       INTEGER            CONSTS
  113   KD6        INTEGER              CONSTS    101   KE        INTEGER            CONSTS
  100   KI         INTEGER              CONSTS  36531   KODE      INTEGER   ARRAY    BK1
32157   LPB        INTEGER                      32154   LPB0      INTEGER
   77   LQR        INTEGER              CONSTS  32134   MANY      INTEGER
32160   MATYP      INTEGER                      12743   MCEKOD    INTEGER   ARRAY    IN9
 4543   MCKOD      INTEGER   ARRAY      IN9         0   MKOD      INTEGER   ARRAY    IN9
    2   NBET       INTEGER              IN4       120   NBLKFM    INTEGER            CONSTS
  116   NBLKX      INTEGER              CONSTS    117   NBLKY     INTEGER            CONSTS
```

```
   50  NBT      INTEGER            CONSTS       47  NBXE     INTEGER            CONSTS
    1  NBXI     INTEGER            IN4         102  NCUT     INTEGER            CONSTS
  114  NELMX    INTEGER            CONSTS      115  NELMY    INTEGER            CONSTS
40227  NGOB     INTEGER   ARRAY    BK1           4  NGRID    INTEGER            IN7
   27  NMATS    INTEGER            IN1         104  NONE     INTEGER            CONSTS
    4  NOSP     INTEGER            IN4       32153  NP       INTEGER
    2  NRUN     INTEGER            IN7          26  NSEAM    INTEGER            IN1
    3  NSYM     INTEGER            IN4          52  NTR      INTEGER            CONSTS
    2  OFSANG   REAL               IN6           0  ORF      REAL               IN7
  124  PHI      REAL      ARRAY    IN2         214  PHI1     REAL      ARRAY    IN2
    0  PI       REAL               CONSTS        0  PO1      REAL      ARRAY    BK2
    4  PO2      REAL      ARRAY    BK2          10  PO3      REAL      ARRAY    BK2
   17  P1ET     REAL               BK2          14  P1XI     REAL               BK2
   20  P2ET     REAL               BK2          15  P2XI     REAL               BK2
   21  P3ET     REAL               BK2          16  P3XI     REAL               BK2
   46  RLIM     REAL               CONSTS        0  SIG1     REAL      ARRAY    AK2
32170  SIG1BL   REAL                            31  SIG2     REAL      ARRAY    AK2
32171  SIG2BL   REAL                            62  SIG3     REAL      ARRAY    AK2
32172  SIG3BL   REAL                             0  STRANG   REAL               IN6
    0  S13PB    REAL               AK1           6  S13PE    REAL               AK1
    0  S13U     REAL      ARRAY    BK4           1  S23PB    REAL               AK1
    7  S23PE    REAL               AK1        6344  S23U     REAL      ARRAY    BK4
23254  S23V     REAL      ARRAY    BK4           2  S33PB    REAL               AK1
   10  S33PE    REAL               AK1       14710  S33U     REAL      ARRAY    BK4
          PROGRAM PMLBM 74/855 OPT=2 FTN 4.8+587  86/02/05. 13.56.15  PAGE 8
VARIABLES       TYPE          RELOCATION
31620  S33V     REAL      ARRAY    BK4       40164  S33W     REAL      ARRAY    BK4
   14  THIKNS   REAL      ARRAY    IN5       32136  TIM      REAL
32133  TIMB     REAL                         32135  TIME     REAL
    0  TITLE    REAL      ARRAY    IN1         226  UNEG     REAL      ARRAY    AK2
32165  UNEGBL   REAL                             3  UPB      REAL               AK1
   11  UPE      REAL               AK1         113  UPOS     REAL      ARRAY    AK2
32162  UPOSBL   REAL                            24  V        REAL               IN1
  257  VNEG     REAL      ARRAY    AK2       32166  VNEGBL   REAL
    4  VPB      REAL               AK1          12  VPE      REAL               AK1
```

```
  144  VPOS      REAL      ARRAY    AK2        32163  VPOSBL   REAL
    2  V1        REAL               CONSTS         3  V2       REAL               CONSTS
  310  WNEG      REAL      ARRAY    AK2        32167  WNEGBL   REAL
    5  WPB       REAL               AK1           13  WPE      REAL               AK1
  175  WPOS      REAL      ARRAY    AK2        32164  WPOSBL   REAL
    0  XO        REAL      ARRAY    IN5            4  YO       REAL      ARRAY    IN5
   10  ZO        REAL      ARRAY    IN5

FILE NAMES        MODE
 4130  COFFS               12334  COFFSD              24670  CSEAM1               26744  CSEAM2
14410  DISC                    0  INPUT                2054  OUTPUT   FMT         20540  SEAM1
22614  SEAM2               14410  TAPE10              16464  TAPE12                6204  TAPE13
 4130  TAPE16              12334  TAPE20              10260  TAPE30   UNFMT       20540  TAPE31
22614  TAPE32              24670  TAPE41              26744  TAPE42

EXTERNALS         TYPE   ARGS
       CLOSMS              1                    CONSTN            0
       INFCOF              0                    INFILE            0
       INIGOB              0                    OFSIM             1
       PRTINP              0                    PSTRES            0
       RDEDIT              1                    SECOND            1
       SOLVER             12                    WRITMS            5

STATEMENT LABELS
    0  100                       31260  200                      0  300
    0  400                       31136  420                      0  500
31326  999                       31616  10000  FMT           31631  10010  FMT
31653  10020  FMT                31666  10030  FMT           31701  10040  FMT
31710  10050  FMT                31740  10055  FMT           31767  10060  FMT
31776  10065  FMT                32006  10070  FMT           32032  10080  FMT
32045  10090  FMT                32061  10100  FMT           32074  10200  FMT
32117  10300  FMT                31444  27000  FMT

LOOPS  LABEL   INDEX   FROM-TO   LENGTH   PROPERTIES
31117  100     I        131 135      3B       OPT
```

```
31221  500      NP          226 255      73B               EXT REFS  NOT INNER
31231  400      JP          229 254      60B               EXT REFS  NOT INNER
31237  300      IP          230 253      45B               EXT REFS

COMMON BLOCKS   LENGTH
       BK1       19735
       BK2          18
       BK3          11
       BK4       20124
       BK5       26400
       CONSTS       81
       IN1          24
       IN2         196
          PROGRAM PMLBM 74/855 OPT=2 FTN 4.8+587  86/02/05. 13.56.15  PAGE 9
COMMON BLOCKS   LENGTH
       IN3          12
       IN4           9
       IN4A         11
       IN5          16
       IN6           3
       IN7           5
       IN9        8803
       AK1          12
       AK2         225

STATISTICS
  PROGRAM LENGTH                 2566B    1398
  BUFFER LENGTH                 27406B   12038
  CM LABELED COMMON LENGTH     223645B   75685
          60000B CM USED
          SUBROUTINE BLINFL 74/855 OPT=2 FTN 4.8+587  86/02/05. 13.56.15  PAGE 1

     1              SUBROUTINE BLINFL (IP, JP, NP, INDB1, INDB2, INDB3, THICK, ERRORB
                   #                    , ITER, COARSE, JB, JE, IB, IE )
               C
```

```
C..THIS SUBROUTINE WILL CALCULATE THE AVERAGE INFLUENCE OF EACH BLOCK
C  NOT IN THE NEIGHBORHOOD OF THE BLOCK BEING PROCESSED TO THE BLOCK
C  BEING PROCESSED.  BLOCK TO BLOCK COEFFICIENTS ARE USED.
C
      COMMON/BK1/    DUB(3200), DVB(3200), DWB(3200),DWBI(3200),
     #               DE(2100),  INDEX(805),  KODE(25), INDX13(805)
     #               ,NGOB(3200)
      COMMON/BK2/    PO1(4), PO2(4), PO3(4),
     #               P1XI,   P2XI,   P3XI,
     #               P1ET,   P2ET,   P3ET
      COMMON/BK4/    S13U(3300), S23U(3300), S33U(3300),
     #               S23V(3300), S33V(3300), S33W(3300),
     #               CF(3,27,4)
      COMMON/BK5/    D1U(3300), D2U(3300), D3U(3300),
     #               D2V(3300), D3V(3300),
     #               D1W(3300), D2W(3300), D3W(3300)
      COMMON/CONSTS/ PI, FACT, V1, V2, CONS, COND, CON, IAMAT(28),
     #               HEW, HBW, EW, RLIM, NBXE, NBT, KB, NTR, ITMINI,
     #               ITMAXI, IPHASE, JOG, ID(4,4), LQR, KI, KE, NCUT,
     #               ACC, NONE, KD0, KD1, KD2, KD3, KD4, KD5, KD6
     #               , NELMX, NELMY, NBLKX, NBLKY, NBLKFM
      COMMON/IN1/    TITLE(20), V, E, NSEAM, NMATS
      COMMON/IN2/    ES(28), GS(28), COHES(28),
     #               PHI(28), COHES1(28), PHI1(28), IGOB(28)
      COMMON/IN4/    BW, NBXI, NBET, NSYM, NOSP, IFXS, IFXE, IFYS, IFYE
      COMMON/IN4A/   IEX, EXR(10)
      COMMON/IN9/    MKOD(2403), MCKOD(3200), MCEKOD(3200)
      COMMON/AK1/    S13PB, S23PB, S33PB, UPB, VPB, WPB,
     #               S13PE, S23PE, S33PE, UPE, VPE, WPE
C
      LOGICAL COARSE
C..
C..CALCULATE THE INDEX TO THE CURRENT BLOCK
C
      LPBI = (NP-1) * NBXE + (JP-1) * NBET + IP
```

```
      C
      C..CONSIDER THE INFLUENCE OF ALL BLOCKS TO THE BLOCK
      C  BEING PROCESSED
      C
      C
         50 DO 400 NS = 1,NSEAM
      C
      C..SET SAME SEAM OR DIFFERENT SEAM FLAGS
      C
              ISAME = 0
              IDIFF = 0
              ILOWER = 0
              IF(NS .EQ. NP) ISAME = 1
              IF(NS .NE. NP) IDIFF = 1
              IF(NS .LT. NP) ILOWER = 1

      C
              LSBO = (NS - 1) * NBXE
      C
SUBROUTINE BLINFL 74/855 OPT=2 FTN 4.8+587  86/02/05. 13.56.15  PAGE 2

      C..CALCULATE THE INDEX TO THE BLOCK ABOVE
      C  OR BELOW THE CURRENT BLOCK
      C
              LXB = (NS-1) * NBXE + (JP-1) * NBET + IP
      C
              INDIC = ID(NP,NS)
              NUMO = IABS(INDIC)
              NPS = NS - NP
              PZT = FLOAT(ISIGN (NONE, NPS))
      C
              DO 300 JS = 1, NBXI
                JPS = JP - JS
                PXI = FLOAT(ISIGN(NONE, JPS))
                JPS = IABS(JPS)
```

```
C
              DO 200 IS = 1,NBET
                IPS = IP - IS
                PET = FLOAT( ISIGN( NONE, IPS))
                IPS = IABS(IPS)
                PXE = PXI * PET
                PXZ = PXI * PZT
                PEZ = PET * PZT
                PXEZ = PXE * PZT
                NUM = NUMO + JPS * 40 + IPS
                LSB = LSBO + (JS - 1) * NBET + IS
C
C..CHECK FOR COARSE MESH BLOCK
C
                IF( COARSE) GO TO 100
C
C
C..IF THE FOLLOWING VALUE IS NOT NEGATIVE THEN WE HAVE
C  BLOCK TO BLOCK COEFFICIENTS
C
                IF (ID(NP,NS) .GT. 0) GO TO 100
C
C..SKIP BLOCK IF BLOCK BEING PROCESSED IS A FINE MESH
C  BLOCK AND THE INFLUENCING BLOCK IS IN THE NEIGHBORHOOD
C
                  IF (JS .GE. JB .AND. JS .LE. JE
     #               .AND. IS .GE. IB .AND. IS .LE. IE ) GO TO 200
C
C
C..STRESS COMPONENTS
C
  100             S13PB = S13PB + S13U(NUM) * DUB(LSB)
     #                         + S23U(NUM) * DVB(LSB) * PXE
     #                         + S33U(NUM) * PXZ *
     #             (DWB(LSB) - ISAME * DWBI(LPBI) - IDIFF * DWBI(LXB))
```

```
                  S23PB = S23PB + S23U(NUM) * DUB(LSB) * PXE
         #                       + S23V(NUM) * DVB(LSB)
         #                       + S33V(NUM) * PEZ *
         #         (DWB(LSB) - ISAME * DWBI(LPBI) - IDIFF * DWBI(LXB))
                  S33PB = S33PB + S33U(NUM) * DUB(LSB) * PXZ
         #                       + S33V(NUM) * DVB(LSB) * PEZ
         #                       + S33W(NUM) *
         #         (DWB(LSB) - ISAME * DWBI(LPBI) - IDIFF * DWBI(LXB))
SUBROUTINE BLINFL 74/855 OPT=2 FTN 4.8+587  86/02/05. 13.56.15  PAGE 3

                 IF (IPHASE .EQ. 6HINPUT ) GO TO 200
      C
      C..DEFINE REVISED BLOCK CLOSURES
      C
                  IF(IDIFF .EQ. 1) GO TO 120
      C
      C..INFLUENCE OF BLOCK ON ITSELF
      C
                  RDWB = DWB(LSB) - DWBI(LPBI)
                  IF(LSB.EQ.LPBI.AND.RDWB.LT.0
         #            .AND. NGOB(LSB) .GT. 0)
         #          RDWB = 0
                    GO TO 150
        120         RDWB = DWB(LSB)
                    IF(NGOB(LSB) .EQ. 0) GO TO 150
                    IF(ILOWER .EQ. 1)    GO TO 140
      C
      C..INFLUENCE OF GOB IN A HIGHER SEAM
      C
                    RDWB = DWB(LSB) - DWBI(LSB)
                    IF(RDWB .LT. 0) RDWB = 0
                    GO TO 150
      C
      C..INFLUENCE OF GOB IN A LOWER SEAM
      C
```

```
      140           IF(DWB(LSB) .LT. DWBI(LSB)) RDWB = 2 * DWB(LSB)
C..RDWB EQUALS 2 * DWBI + (DWB-DWBI) IF DWB > DWBI
                   IF(DWB(LSB) .GE. DWBI(LSB)) RDWB = DWB(LSB) + DWBI(LSB)
      150           CONTINUE
C
C
C..DISPLACEMENT COMPONENTS
C
                   UPB = UPB + D1U(NUM) * DUB(LSB) * PZT
     #                       + D2U(NUM) * DVB(LSB) * PXEZ
     #                       + D1W(NUM) * PXI * DWB(LSB)
                   VPB = VPB + D2U(NUM) * DUB(LSB) * PXEZ
     #                       + D2V(NUM) * DVB(LSB) * PZT
     #                       + D2W(NUM) * PET * DWB(LSB)
                  TEMP =  D3U(NUM) * DUB(LSB) * PXI
     #                       + D3V(NUM) * DVB(LSB) * PET
     #                       + D3W(NUM) * PZT * RDWB
                    IF(IDIFF .EQ. 1) TEMP = -TEMP
                    WPB = WPB + TEMP
C
C
  200     CONTINUE
  300   CONTINUE
  400 CONTINUE
C
C..IF NOT A COARSE MESH BLOCK RETURN
C
      IF( IP .GE. IFYS .AND. IP .LE. IFYE .AND.
     #    JP .GE. IFXS .AND. JP .LE. IFXE)      RETURN
C
C..ACCESS MATERIAL PROPERTIES FOR THE BLOCK BEING PROCESSED
C
```

SUBROUTINE BLINFL 74/855 OPT=2 FTN 4.8+587 86/02/05. 13.56.15 PAGE 4

```
      IMAT = MCKOD(LPBI)
```

```
C
C..CHECK TO SEE IF THIS IS AN OPENING
C
      IEXT = MCEKOD(LPBI)
      STFS = -GS(IMAT) / THICK * (1. - EXR( IEXT ) )
      STFN = -ES(IMAT) / THICK * (1. - EXR( IEXT ) )
C
C..CALCULATE PRIMITIVE STRESS COMPONENTS
C
      JLP = ( JP - 1 ) * 5 + 2
      IKP = ( IP - 1 ) * 5 + 2
      P1  = PO1(NP) + JLP * P1XI + IKP * P1ET
      P2  = PO2(NP) + JLP * P2XI + IKP * P2ET
      P3  = PO3(NP) + JLP * P3XI + IKP * P3ET
C
C
C
          T1 = S13PB * CONS
          T2 = S23PB * CONS
          T3 = S33PB * CONS
C
C.. FIX FOR BLOCK CORRECTION FACTORS
C
          SS1 = S13U(1)
          SS2 = S23V(1)
          SS3 = S33W(1)
          CC  = 1. - EXR(IEXT)
          GG  = -GS(IMAT) * CC
          EE  = -ES(IMAT) * CC
          CFB1 = CON / ( SS1 + ( GG / THICK / CONS ) )
          CFB2 = CON / ( SS2 + ( GG / THICK / CONS ) )
          CFB3 = CON / ( SS3 + ( EE / THICK / CONS ) )
C
C.. END FIX
C
```

```
      C
      C..OBTAIN THE STRESS RESIDUES AT THIS BLOCK
      C
            KOD = IMAT - 27
            IF( KOD ) 1100,1000,4000
      C
      C..MINED OUT BLOCK
      C
       1000     S13 = P1 + T1
                S23 = P2 + T2
                S33 = P3 + T3
                GOTO 2100
      C
      C..UNMINED AND DEFORMABLE BLOCK
      C
       1100     S13 = STFS * DUB(LPBI) + T1
                S23 = STFS * DVB(LPBI) + T2
                S33 = STFN * ( DWB(LPBI) - DWBI(LPBI) ) + T3
      C
                GO TO 2200
      C
```

SUBROUTINE BLINFL 74/855 OPT=2 FTN 4.8+587 86/02/05. 13.56.15 PAGE 5

```
      C..OBTAIN THE RIDE AND CLOSURE CORRECTIONS
      C  CHECK FOR THE COMPLETE CLOSURE AT THIS BLOCK
      C  IF YES, SET THE CLOSURE CORRECTION ACCORDINGLY
      C
       2100     DELU = S13 * CON/SS1
                DELV = S23 * CON/SS2
                DELW = S33 * CON/SS3
                GO TO 2300
       2200     DELU = S13 * CFB1
                DELV = S23 * CFB2
                DELW = S33 * CFB3
      C
```

```
C
C..OBTAIN THE ERRORS IN THE RIDE AND CLOSURE VALUES
C
 2300     ERORl = ABS(DELU)
          EROR2 = ABS(DELV)
          EROR3 = ABS(DELW)
          IF (ERORl .GT. ACC) INDBl = INDBl + 1
          IF (EROR2 .GT. ACC) INDB2 = INDB2 + 1
          IF (EROR3 .GT. ACC) INDB3 = INDB3 + 1
          ERRORB = AMAX1(ERRORB, ERORl, EROR2, EROR3)
C
C..IF THIS IS THE OUTPUT CYCLE, OBTAIN THE TOTAL STRESSES AND
C  DISPLACEMENTS AT THIS BLOCK
C
C
          IF (IPHASE .EQ. 6HINPUT ) GO TO 2400
            SIGlBL = Pl + Tl
            SIG2BL = P2 + T2
            SIG3BL = P3 + T3
            UPOSBL = UPB * COND
            VPOSBL =  VPB * COND
            WNEGBL = -1 * WPB * COND
C
C..FOR PARTIALLY EXTRACTED BLOCKS DISTRIBUTE THE
C  INITIAL CLOSURE EQUALLY BETWEEN FLOOR AND ROOF
C
            IF( IGOB(IMAT) .EQ. 0 ) WNEGBL=WNEGBL + .5 * DWBI(LPBI)
C
            UNEGBL = UPOSBL + DUB(LPBI)
            VNEGBL = VPOSBL + DVB(LPBI)
            WPOSBL = WNEGBL - DWB(LPBI)
C
C..WRITE OUT TOTAL STRESSES FOR COARSE BLOCK
C
          WRITE(30)   UPOSBL, VPOSBL, WPOSBL
```

```
      #                  , UNEGBL, VNEGBL, WNEGBL
      #                  , SIG1BL, SIG2BL, SIG3BL
C
            GO TO 4000
C
C..OBTAIN THE CORRECTED VALUES OF RIDES AND CLOSURES
C
 2400         DUB(LPBI) = DUB(LPBI) + DELU
              DVB(LPBI) = DVB(LPBI) + DELV
              DWB(LPBI) = DWB(LPBI) + DELW
SUBROUTINE BLINFL 74/855 OPT=2 FTN 4.8+587  86/02/05. 13.56.15  PAGE 6

              IF( DWB(LPBI) .LT. 0 .AND. IGOB(IMAT) .EQ. 1 )
      #          DWB(LPBI) = 0.0
              IF( DWB(LPBI) .GT. THICK ) DWB(LBPI) = THICK
C
 4000   CONTINUE
      RETURN
      END


   SYMBOLIC REFERENCE MAP (R=1)

ENTRY POINTS
    3  BLINFL

VARIABLES         TYPE           RELOCATION
  103  ACC        REAL               CONSTS           0  BW        REAL                IN4
  604  CC         REAL                            46530  CF        REAL       ARRAY    BK4
  607  CFB1       REAL                              610  CFB2      REAL
  611  CFB3       REAL                                0  COARSE    LOGICAL              F.P.
   70  COHES      REAL        ARRAY  IN2            160  COHES1    REAL       ARRAY    IN2
    6  CON        REAL               CONSTS           5  COND      REAL                CONSTS
    4  CONS       REAL               CONSTS       31000  DE        REAL       ARRAY    BK1
  616  DELU       REAL                              617  DELV      REAL
```

620	DELW	REAL			0	DUB	REAL	ARRAY	BK1
6200	DVB	REAL	ARRAY	BK1	14400	DWB	REAL	ARRAY	BK1
22600	DWBI	REAL	ARRAY	BK1	0	D1U	REAL	ARRAY	BK5
40164	D1W	REAL	ARRAY	BK5	6344	D2U	REAL	ARRAY	BK5
23254	D2V	REAL	ARRAY	BK5	46530	D2W	REAL	ARRAY	BK5
14710	D3U	REAL	ARRAY	BK5	31620	D3V	REAL	ARRAY	BK5
55074	D3W	REAL	ARRAY	BK5	25	E	REAL		IN1
606	EE	REAL			621	EROR1	REAL		
622	EROR2	REAL			623	EROR3	REAL		
0	ERRORB	REAL		F.P.	0	ES	REAL	ARRAY	IN2
45	EW	REAL		CONSTS	1	EXR	REAL	ARRAY	IN4A
1	FACT	REAL		CONSTS	605	GG	REAL		
34	GS	REAL	ARRAY	IN2	44	HBW	REAL		CONSTS
43	HEW	REAL		CONSTS	7	IAMAT	INTEGER	ARRAY	CONSTS
0	IB	INTEGER		F.P.	57	ID	INTEGER	ARRAY	CONSTS
537	IDIFF	INTEGER			0	IE	INTEGER		F.P.
0	IEX	INTEGER		IN4A	566	IEXT	INTEGER		
6	IFXE	INTEGER		IN4	5	IFXS	INTEGER		IN4
10	IFYE	INTEGER		IN4	7	IFYS	INTEGER		IN4
250	IGOB	INTEGER	ARRAY	IN2	572	IKP	INTEGER		
540	ILOWER	INTEGER			565	IMAT	INTEGER		
0	INDB1	INTEGER		F.P.	0	INDB2	INTEGER		F.P.
0	INDB3	INTEGER		F.P.	35064	INDEX	INTEGER	ARRAY	BK1
543	INDIC	INTEGER			36562	INDX13	INTEGER	ARRAY	BK1
0	IP	INTEGER		F.P.	55	IPHASE	INTEGER		CONSTS
553	IPS	INTEGER			552	IS	INTEGER		
536	ISAME	INTEGER			0	ITER	INTEGER	*UNUSED	F.P.
54	ITMAXI	INTEGER		CONSTS	53	ITMINI	INTEGER		CONSTS
0	JB	INTEGER		F.P.	0	JE	INTEGER		F.P.
571	JLP	INTEGER			56	JOG	INTEGER		CONSTS
0	JP	INTEGER		F.P.	550	JPS	INTEGER		

SUBROUTINE BLINFL 74/855 OPT=2 FTN 4.8+587 86/02/05. 13.56.15 PAGE 7

VARIABLES		TYPE		RELOCATION					
547	JS	INTEGER			51	KB	INTEGER		CONSTS
105	KD0	INTEGER		CONSTS	106	KD1	INTEGER		CONSTS

107	KD2	INTEGER		CONSTS	110	KD3	INTEGER		CONSTS
111	KD4	INTEGER		CONSTS	112	KD5	INTEGER		CONSTS
113	KD6	INTEGER		CONSTS	101	KE	INTEGER		CONSTS
100	KI	INTEGER		CONSTS	612	KOD	INTEGER		
36531	KODE	INTEGER	ARRAY	BK1	635	LBPI	INTEGER	*UNDEF	
534	LPBI	INTEGER			77	LQR	INTEGER		CONSTS
562	LSB	INTEGER			541	LSBO	INTEGER		
542	LXB	INTEGER			12743	MCEKOD	INTEGER	ARRAY	IN9
4543	MCKOD	INTEGER	ARRAY	IN9	0	MKOD	INTEGER	ARRAY	IN9
2	NBET	INTEGER		IN4	120	NBLKFM	INTEGER		CONSTS
116	NBLKX	INTEGER		CONSTS	117	NBLKY	INTEGER		CONSTS
50	NBT	INTEGER		CONSTS	47	NBXE	INTEGER		CONSTS
1	NBXI	INTEGER		IN4	102	NCUT	INTEGER		CONSTS
114	NELMX	INTEGER		CONSTS	115	NELMY	INTEGER		CONSTS
40227	NGOB	INTEGER	ARRAY	BK1	27	NMATS	INTEGER		IN1
104	NONE	INTEGER		CONSTS	4	NOSP	INTEGER		IN4
0	NP	INTEGER		F.P.	545	NPS	INTEGER		
535	NS	INTEGER			26	NSEAM	INTEGER		IN1
3	NSYM	INTEGER		IN4	52	NTR	INTEGER		CONSTS
561	NUM	INTEGER			544	NUMO	INTEGER		
554	PET	REAL			557	PEZ	REAL		
124	PHI	REAL	ARRAY	IN2	214	PHI1	REAL	ARRAY	IN2
0	PI	REAL		CONSTS	0	PO1	REAL	ARRAY	BK2
4	PO2	REAL	ARRAY	BK2	10	PO3	REAL	ARRAY	BK2
555	PXE	REAL			560	PXEZ	REAL		
551	PXI	REAL			556	PXZ	REAL		
546	PZT	REAL			573	P1	REAL		
17	P1ET	REAL		BK2	14	P1XI	REAL		BK2
574	P2	REAL			20	P2ET	REAL		BK2
15	P2XI	REAL		BK2	575	P3	REAL		
21	P3ET	REAL		BK2	16	P3XI	REAL		BK2
563	RDWB	REAL			46	RLIM	REAL		CONSTS
624	SIG1BL	REAL			625	SIG2BL	REAL		
626	SIG3BL	REAL			601	SS1	REAL		
602	SS2	REAL			603	SS3	REAL		

```
  570  STFN      REAL                            567  STFS     REAL
  613  S13       REAL                              0  S13PB    REAL              AK1
    6  S13PE     REAL               AK1            0  S13U     REAL     ARRAY    BK4
  614  S23       REAL                              1  S23PB    REAL              AK1
    7  S23PE     REAL               AK1         6344  S23U     REAL     ARRAY    BK4
23254  S23V      REAL      ARRAY    BK4          615  S33      REAL
    2  S33PB     REAL               AK1           10  S33PE    REAL              AK1
14710  S33U      REAL      ARRAY    BK4        31620  S33V     REAL     ARRAY    BK4
40164  S33W      REAL      ARRAY    BK4          564  TEMP     REAL
    0  THICK     REAL                F.P.          0  TITLE    REAL     ARRAY    IN1
  576  T1        REAL                            577  T2       REAL
  600  T3        REAL                            632  UNEGBL   REAL
    3  UPB       REAL               AK1           11  UPE      REAL              AK1
  627  UPOSBL    REAL                             24  V        REAL              IN1
  633  VNEGBL    REAL                              4  VPB      REAL              AK1
   12  VPE       REAL               AK1          630  VPOSBL   REAL
    2  V1        REAL               CONSTS         3  V2       REAL              CONSTS
  631  WNEGBL    REAL                              5  WPB      REAL              AK1
   13  WPE       REAL               AK1          634  WPOSBL   REAL
          SUBROUTINE BLINFL 74/855 OPT=2 FTN 4.8+587  86/02/05. 13.56.15  PAGE 8
FILE NAMES       MODE
       TAPE30    UNFMT
 INLINE FUNCTIONS  TYPE   ARGS
       ABS       REAL       1  INTRIN                AMAX1    REAL      0  INTRIN
       FLOAT     REAL       1  INTRIN                IABS     INTEGER   1  INTRIN
       ISIGN     INTEGER    2  INTRIN

 STATEMENT LABELS
    0  50          INACTIVE         114  100                               162  120
  171  140                          201  150                               236  200
    0  300                            0  400                                 0  1000        INACTIVE
  346  1100                         361  2100                              371  2200
  377  2300                         463  2400                              501  4000

 LOOPS  LABEL    INDEX     FROM-TO     LENGTH     PROPERTIES
```

```
  21  400     NS          44 163     226B                NOT INNER
  56  300     JS          68 162     165B                NOT INNER
  74  200     IS          73 161     144B         OPT

COMMON BLOCKS   LENGTH
       BK1       19735
       BK2          18
       BK4       20124
       BK5       26400
       CONSTS       81
       IN1          24
       IN2         196
       IN4           9
       IN4A         11
       IN9        8803
       AK1          12

STATISTICS
  PROGRAM LENGTH                   653B     427
  CM LABELED COMMON LENGTH      223225B   75413
        60000B CM USED
        BLOCK DATA BLKDAT.  74/855 OPT=2 FTN 4.8+587  86/02/05. 13.56.15  PAGE 1

          C
          C   INITIALIZE ARRAY OF ACCEPTABLE PROPERTY CODES
          C
                BLOCK DATA
          C
                COMMON/CONSTS/ PI, FACT, V1, V2, CONS, COND, CON, IAMAT(28),
               #               HEW, HBW, EW, RLIM, NBXE, NBT, KB, NTR, ITMINI,
               #               ITMAXI, IPHASE, JOG, ID(4,4), LQR, KI, KE, NCUT,
               #               ACC, NONE, KD0, KD1, KD2, KD3, KD4, KD5, KD6
               #               , NELMX, NELMY, NBLKX, NBLKY, NBLKFM
          C
                DATA IAMAT/   "1", "2", "A", "B", "C", "D", "E", "F", "G",
```

```
                #               "H", "I", "J", "K", "L", "M", "N", "O", "P",
                #               "Q", "R", "S", "T", "U", "V", "W", "X", "Y", "Z" /
   15        C
                 END


      SYMBOLIC REFERENCE MAP (R=1)

VARIABLES        TYPE           RELOCATION
  103  ACC       REAL                CONSTS         6  CON       REAL                CONSTS
    5  COND      REAL                CONSTS         4  CONS      REAL                CONSTS
   45  EW        REAL                CONSTS         1  FACT      REAL                CONSTS
   44  HBW       REAL                CONSTS        43  HEW       REAL                CONSTS
    7  IAMAT     INTEGER    ARRAY    CONSTS        57  ID        INTEGER    ARRAY    CONSTS
   55  IPHASE    INTEGER             CONSTS        54  ITMAXI    INTEGER             CONSTS
   53  ITMINI    INTEGER             CONSTS        56  JOG       INTEGER             CONSTS
   51  KB        INTEGER             CONSTS       105  KD0       INTEGER             CONSTS
  106  KD1       INTEGER             CONSTS       107  KD2       INTEGER             CONSTS
  110  KD3       INTEGER             CONSTS       111  KD4       INTEGER             CONSTS
  112  KD5       INTEGER             CONSTS       113  KD6       INTEGER             CONSTS
  101  KE        INTEGER             CONSTS       100  KI        INTEGER             CONSTS
   77  LQR       INTEGER             CONSTS       120  NBLKFM    INTEGER             CONSTS
  116  NBLKX     INTEGER             CONSTS       117  NBLKY     INTEGER             CONSTS
   50  NBT       INTEGER             CONSTS        47  NBXE      INTEGER             CONSTS
  102  NCUT      INTEGER             CONSTS       114  NELMX     INTEGER             CONSTS
  115  NELMY     INTEGER             CONSTS       104  NONE      INTEGER             CONSTS
   52  NTR       INTEGER             CONSTS         0  PI        REAL                CONSTS
   46  RLIM      REAL                CONSTS         2  V1        REAL                CONSTS
    3  V2        REAL                CONSTS

COMMON BLOCKS   LENGTH
      CONSTS        81

STATISTICS
  PROGRAM LENGTH                 0B        0
```

```
CM LABELED COMMON LENGTH       121B       81
      60000B CM USED
      SUBROUTINE BLUPDT 74/875 OPT=2 FTN 4.8+670  87/09/01. 13.42.10  PAGE 1

      SUBROUTINE BLUPDT (IP,JP,NP,LPR,LPBO,LPE,LPB,LPBFM,LPBOFM)
C
C..THIS SUBROUTINE WILL UPDATE THE ELEMENTAL RIDE AND CLOSURE
C  VALUES FOR ALL OTHER SYMMETRICAL BLOCKS ON THE SAME SEAM
C  THE MASS STORAGE FILE WILL THEN BE UPDATED TO REFLECT THE
C  CHANGES.  THE AVERAGE BLOCK VALUES FOR THE SYMMETRICAL BLOCK
C  WILL BE UPDATED IN CORE ALSO.
C
      COMMON/BK1/    DUB(3200), DVB(3200), DWB(3200),DWBI(3200),
     #               DE(2100),  INDEX(805),  KODE(25), INDX13(805)
     #               ,NGOB(3200)
      COMMON/CONSTS/ PI, FACT, V1, V2, CONS, COND, CON, IAMAT(28),
     #               HEW, HBW, EW, RLIM, NBXE, NBT, KB, NTR, ITMINI,
     #               ITMAXI, IPHASE, JOG, ID(4,4), LQR, KI, KE, NCUT,
     #               ACC, NONE, KD0, KD1, KD2, KD3, KD4, KD5, KD6
     #               , NELMX, NELMY, NBLKX, NBLKY, NBLKFM
      COMMON/IN4/    BW, NBXI, NBET, NSYM, NOSP, IFXS, IFXE, IFYS, IFYE
      COMMON/IN4A/   IEX, EXR(10)
C
C..NSYM = 1 FOR NO SYMMETRY
C         2 FOR COLUMN SYMMETRY
C         3 FOR ROW SYMMETRY
C         4 FOR BOTH ROW AND COLUMN SYMMETRY
C
      GO TO (1000, 200, 300, 100), NSYM
C
  100 JOG = 2
C
C..SETUP VARIABLES FOR COLUMN SYMMETRY
C
  200 JP1 = (NBXI + 1) - JP
```

```
      LQB = LPBO + (JP1 - 1) * NBET + IP
      LQBFM = LPBOFM + (JP1 - IFXS) * NBLKY + IP - IFYS + 1
      IU  = -1
      IV  =  1
      JL  =  6
      IK  =  0
      GO TO 500
C
C..SETUP VARIABLES FOR ROW SYMMETRY
C
  300 IP1 = (NBET + 1) - IP
      LQB = LPBO + (JP - 1) * NBET + IP1
      LQBFM = LPBOFM + (JP - IFXS) * NBLKY + IP1 - IFYS + 1
      IU  =  1
      IV  = -1
      JL  =  0
      IK  =  6
      GO TO 500
C
C..SETUP VARIABLES FOR BOTH COLUMN AND ROW SYMMETRY
C
  400 LQB = LPBO + (JP1-1) * NBET + IP1
      LQBFM = LPBOFM + (JP1 - IFXS) * NBLKY + IP1 - IFYS + 1
      IU  = -1
      IV  = -1
      JL  =  6
SUBROUTINE BLUPDT 74/875 OPT=2 FTN 4.8+670  87/09/01. 13.42.10  PAGE 2

      IK  =  6
C
C..STORE ELEMENTAL RIDES AND CLOSURES SYMMETRICALLY IN PSEUDO BLOCK
C  NUMBER 28 TEMPORARILY PRIOR TO WRITING OUT TO ACTUAL RECORD
C  CONTAINING THE SYMMETRICAL BLOCKS DATA.
C
C
```

```
C..IF COARSE MESH BLOCK SKIP TO THE UPDATING OF BLOCK VALUES.
C
  500 IF( IP .LT. IFYS .OR. IP .GT. IFYE ) GO TO 750
      IF( JP .LT. IFXS .OR. JP .GT. IFXE ) GO TO 750
      N = 0
      NSB = LQBFM + 4
      DO 700 LP = 1,5
        DO 600 KP = 1,5
      LPE = LPR + N
      LQE = LQR + ((LP - JL) * IU - 1) * 5 + (KP - IK) * IV - 1
          N = N + 1
          DE(LQE) = DE(LPE) * IU
          DE(LQE+25) = DE(LPE+25) * IV
          DE(LQE+50) = DE(LPE+50)
  600   CONTINUE
  700 CONTINUE
C
C..WRITE PSEUDO BLOCK 28 ELEMENTAL VALUES TO THE SYMMETRICAL
C  BLOCKS RECORD ON DISK
C
      CALL WRITMS (10, DE(LQR), 75, NSB, 1)
C
C
C..UPDATE AVERAGE BLOCK VALUES
C
  750 DUB(LQB) = DUB(LPB) * IU
      DVB(LQB) = DVB(LPB) * IV
      DWB(LQB) = DWB(LPB)
C
      IF(JOG-2)1000,800,900
  800 JOG = 3
      GO TO 300
  900 JOG = 1
      GO TO 400
C
```

```
    100             1000 RETURN
                         END


       SYMBOLIC REFERENCE MAP (R=1)

ENTRY POINTS
    3  BLUPDT

VARIABLES          TYPE             RELOCATION
  103  ACC        REAL                 CONSTS          0  BW        REAL                 IN4
    6  CON        REAL                 CONSTS          5  COND      REAL                 CONSTS
          SUBROUTINE BLUPDT 74/875 OPT=2 FTN 4.8+670  87/09/01  13.42.10  PAGE 3
VARIABLES          TYPE             RELOCATION
    4  CONS       REAL                 CONSTS      31000  DE        REAL       ARRAY     BK1
    0  DUB        REAL        ARRAY    BK1          6200  DVB       REAL       ARRAY     BK1
14400  DWB        REAL        ARRAY    BK1         22600  DWBI      REAL       ARRAY     BK1
   45  EW         REAL                 CONSTS          1  EXR       REAL       ARRAY     IN4A
    1  FACT       REAL                 CONSTS         44  HBW       REAL                 CONSTS
   43  HEW        REAL                 CONSTS          7  IAMAT     INTEGER    ARRAY     CONSTS
   57  ID         INTEGER     ARRAY    CONSTS          0  IEX       INTEGER              IN4A
    6  IFXE       INTEGER              IN4             5  IFXS      INTEGER              IN4
   10  IFYE       INTEGER              IN4             7  IFYS      INTEGER              IN4
  177  IK         INTEGER                          35064  INDEX     INTEGER    ARRAY     BK1
36562  INDX13     INTEGER     ARRAY    BK1             0  IP        INTEGER               F.P.
   55  IPHASE     INTEGER              CONSTS        200  IP1       INTEGER
   54  ITMAXI     INTEGER              CONSTS         53  ITMINI    INTEGER              CONSTS
  174  IU         INTEGER                            175  IV        INTEGER
  176  JL         INTEGER                             56  JOG       INTEGER              CONSTS
    0  JP         INTEGER               F.P.         171  JP1       INTEGER
   51  KB         INTEGER              CONSTS        105  KD0       INTEGER              CONSTS
  106  KD1        INTEGER              CONSTS        107  KD2       INTEGER              CONSTS
  110  KD3        INTEGER              CONSTS        111  KD4       INTEGER              CONSTS
  112  KD5        INTEGER              CONSTS        113  KD6       INTEGER              CONSTS
  101  KE         INTEGER              CONSTS        100  KI        INTEGER              CONSTS
```

```
36531  KODE     INTEGER   ARRAY    BK1          204  KP      INTEGER
  203  LP       INTEGER                           0  LPB     INTEGER              F.P.
    0  LPBFM    INTEGER  *UNUSED    F.P.          0  LPBO    INTEGER              F.P.
    0  LPBOFM   INTEGER             F.P.          0  LPE     INTEGER              F.P.
    0  LPR      INTEGER             F.P.        172  LQB     INTEGER
  173  LQBFM    INTEGER                         205  LQE     INTEGER
   77  LQR      INTEGER            CONSTS       201  N       INTEGER
    2  NBET     INTEGER            IN4          120  NBLKFM  INTEGER             CONSTS
  116  NBLKX    INTEGER            CONSTS       117  NBLKY   INTEGER             CONSTS
   50  NBT      INTEGER            CONSTS        47  NBXE    INTEGER             CONSTS
    1  NBXI     INTEGER            IN4          102  NCUT    INTEGER             CONSTS
  114  NELMX    INTEGER            CONSTS       115  NELMY   INTEGER             CONSTS
40227  NGOB     INTEGER   ARRAY    BK1          104  NONE    INTEGER             CONSTS
    4  NOSP     INTEGER            IN4            0  NP      INTEGER   *UNUSED    F.P.
  202  NSB      INTEGER                           3  NSYM    INTEGER             IN4
   52  NTR      INTEGER            CONSTS         0  PI      REAL                CONSTS
   46  RLIM     REAL               CONSTS         2  V1      REAL                CONSTS
    3  V2       REAL               CONSTS

EXTERNALS          TYPE   ARGS
       WRITMS               5

STATEMENT LABELS
   17  100                           20  200                               37  300
   56  400                          152  500                                0  600
    0  700                          134  750                                0  800        INACTIVE
  147  900                          151  1000

LOOPS  LABEL    INDEX    FROM-TO    LENGTH   PROPERTIES
  115  700      LP         71 80      14B              NOT INNER
  120  600      KP         72 79       5B     INSTACK

COMMON BLOCKS   LENGTH
       BK1      19735
       CONSTS      81
```

```
      IN4           9
         SUBROUTINE BLUPDT 74/875 OPT=2 FTN 4.8+670  87/09/01. 13.42.10  PAGE 4
COMMON BLOCKS    LENGTH
      IN4A          11

STATISTICS
  PROGRAM LENGTH                  216B     142
  SCM LABELED COMMON LENGTH    46574B   19836
         60000B SCM USED
         SUBROUTINE COEF1 74/855 OPT=2 FTN 4.8+587  86/02/05. 13.56.15  PAGE 1

                  SUBROUTINE COEF1
            C
            C  THIS ROUTINE COMPUTES THE INTRA SEAM INFLUENCE COEFFICIENTS. I.E. THE
            C  INFLUENCES OF AN ELEMENT ON OTHER ELEMENTS AT THE SAME PLANE.
            C
                  COMMON/BK4/    S13U(3300), S23U(3300), S33U(3300),
                 #               S23V(3300), S33V(3300), S33W(3300),
                 #               CF(3,27,4)
                  COMMON/BK5/    D1U(3300), D2U(3300), D3U(3300),
                 #               D2V(3300), D3V(3300),
                 #               D1W(3300), D2W(3300), D3W(3300)
                  COMMON/CONSTS/ PI, FACT, V1, V2, CONS, COND, CON, IAMAT(28),
                 #               HEW, HBW, EW, RLIM, NBXE, NBT, KB, NTR, ITMINI,
                 #               ITMAXI, IPHASE, JOG, ID(4,4), LQR, KI, KE, NCUT,
                 #               ACC, NONE, KD0, KD1, KD2, KD3, KD4, KD5, KD6
                 #               , NELMX, NELMY, NBLKX, NBLKY, NBLKFM
                  COMMON/IN1/    TITLE(20), V, E, NSEAM, NMATS
            C
            C  STATEMENT FUNCTIONS
            C
                  F0(P,R)=ALOG(R-P)
                  F1(P,Q,R)=(P*Q)/((R**2-Q**2)*R)
                  SEG=HBW
                  NUM = 0
```

```
C
      DO 200 J=1,40
        XI=(J-1)*2.*SEG
        XI1=XI-SEG
        XI2=XI+SEG
C
        DO 100 I=1,40
          ET=(I-1)*2.*SEG
          ET1=ET-SEG
          ET2=ET+SEG
          NUM = NUM + 1
          R1=SQRT(XI1*XI1+ET1*ET1)
          R2=SQRT(XI2*XI2+ET2*ET2)
          R3=SQRT(XI1*XI1+ET2*ET2)
          R4=SQRT(XI2*XI2+ET1*ET1)
C
C  DERIVATIVES OF INTEGRAL FUNCTION I--
C
          DERI01=F0(ET2,R3)+F0(ET1,R4)-F0(ET1,R1)-F0(ET2,R2)
          DERI02=F0(XI1,R3)+F0(XI2,R4)-F0(XI1,R1)-F0(XI2,R2)
          DERI03=0.
          IF(ABS(XI).LE.SEG.AND.ABS(ET).LE.SEG) DERI03=-2.*PI
          DERI04=F1(XI1,ET2,R3)+F1(XI2,ET1,R4)-F1(XI1,ET1,R1)
     #          - F1(XI2,ET2,R2)
          DERI05=1./R1+1./R2-1./R3-1./R4
          DERI07=F1(ET2,XI1,R3)+F1(ET1,XI2,R4)-F1(ET1,XI1,R1)
     #         - F1(ET2,XI2,R2)
          DERI09=-DERI04-DERI07
C
C  STRESS COEFFICIENTS
C
          S13U(NUM)=V*DERI04-(1.-V)*DERI09
          S23U(NUM)=V*DERI05
```

```
                  S33U(NUM)=0.0
                  S23V(NUM)=V*DERI07-(1.-V)*DERI09
                  S33V(NUM)=0.0
                  S33W(NUM)=-DERI09
C
C  DISPLACEMENT COEFFICIENTS
C
                  D1U(NUM)=2.*V1*DERI03
                  D2V(NUM)=D1U(NUM)
                  D3W(NUM)=D2V(NUM)
                  D2U(NUM)=0.0
                  D3U(NUM)=V2*DERI01
                  D1W(NUM)=-D3U(NUM)
                  D3V(NUM)=V2*DERI02
                  D2W(NUM)=-D3V(NUM)
  100   CONTINUE
C
  200 CONTINUE
C
      RETURN
      END
```

SYMBOLIC REFERENCE MAP (R=1)

```
ENTRY POINTS
    1   COEF1

VARIABLES          TYPE           RELOCATION
  103   ACC        REAL               CONSTS       46530  CF        REAL        ARRAY     BK4
    6   CON        REAL               CONSTS           5  COND      REAL                  CONSTS
    4   CONS       REAL               CONSTS         236  DERI01    REAL
  237   DERI02     REAL                              240  DERI03    REAL
  241   DERI04     REAL                              242  DERI05    REAL
  243   DERI07     REAL                              244  DERI09    REAL
```

```
     0  D1U        REAL      ARRAY    BK5         40164  D1W      REAL      ARRAY   BK5
  6344  D2U        REAL      ARRAY    BK5         23254  D2V      REAL      ARRAY   BK5
 46530  D2W        REAL      ARRAY    BK5         14710  D3U      REAL      ARRAY   BK5
 31620  D3V        REAL      ARRAY    BK5         55074  D3W      REAL      ARRAY   BK5
    25  E          REAL               IN1           227  ET       REAL
   230  ET1        REAL                             231  ET2      REAL
    45  EW         REAL               CONSTS          1  FACT     REAL              CONSTS
    44  HBW        REAL               CONSTS         43  HEW      REAL              CONSTS
   226  I          INTEGER                            7  IAMAT    INTEGER   ARRAY   CONSTS
    57  ID         INTEGER   ARRAY    CONSTS         55  IPHASE   INTEGER           CONSTS
    54  ITMAXI     INTEGER            CONSTS         53  ITMINI   INTEGER           CONSTS
   222  J          INTEGER                           56  JOG      INTEGER           CONSTS
    51  KB         INTEGER            CONSTS        105  KD0      INTEGER           CONSTS
   106  KD1        INTEGER            CONSTS        107  KD2      INTEGER           CONSTS
   110  KD3        INTEGER            CONSTS        111  KD4      INTEGER           CONSTS
   112  KD5        INTEGER            CONSTS        113  KD6      INTEGER           CONSTS
   101  KE         INTEGER            CONSTS        100  KI       INTEGER           CONSTS
    77  LQR        INTEGER            CONSTS        120  NBLKFM   INTEGER           CONSTS
   116  NBLKX      INTEGER            CONSTS        117  NBLKY    INTEGER           CONSTS
        SUBROUTINE COEF1  74/855 OPT=2 FTN 4.8+587  86/02/05. 13.56.15  PAGE 3
VARIABLES          TYPE          RELOCATION
    50  NBT        INTEGER            CONSTS         47  NBXE     INTEGER           CONSTS
   102  NCUT       INTEGER            CONSTS        114  NELMX    INTEGER           CONSTS
   115  NELMY      INTEGER            CONSTS         27  NMATS    INTEGER           IN1
   104  NONE       INTEGER            CONSTS         26  NSEAM    INTEGER           IN1
    52  NTR        INTEGER            CONSTS        221  NUM      INTEGER
     0  PI         REAL               CONSTS         46  RLIM     REAL              CONSTS
   232  R1         REAL                             233  R2       REAL
   234  R3         REAL                             235  R4       REAL
   220  SEG        REAL                               0  S13U     REAL      ARRAY   BK4
  6344  S23U       REAL      ARRAY    BK4         23254  S23V     REAL      ARRAY   BK4
 14710  S33U       REAL      ARRAY    BK4         31620  S33V     REAL      ARRAY   BK4
 40164  S33W       REAL      ARRAY    BK4             0  TITLE    REAL      ARRAY   IN1
    24  V          REAL               IN1             2  V1       REAL              CONSTS
     3  V2         REAL               CONSTS        223  XI       REAL
```

```
  224  XI1        REAL                                   225  XI2        REAL

EXTERNALS          TYPE   ARGS
       ALOG       REAL      1 LIBRARY                         SQRT       REAL       1 LIBRARY

INLINE FUNCTIONS   TYPE   ARGS
       ABS        REAL      1  INTRIN                         F0         REAL       2    SF
       F1         REAL      3    SF

STATEMENT LABELS
    0  100                                0  200

LOOPS  LABEL    INDEX     FROM-TO    LENGTH    PROPERTIES
   11  200      J           26 75      170B              EXT REFS  NOT INNER
   24  100      I           31 73      153B              EXT REFS

COMMON BLOCKS   LENGTH
       BK4       20124
       BK5       26400
       CONSTS       81
       IN1          24

STATISTICS
  PROGRAM LENGTH                  245B     165
  CM LABELED COMMON LENGTH   133045B   46629
        60000B CM USED
        SUBROUTINE COEF2 74/855 OPT=2 FTN 4.8+587  86/02/05. 13.56.15  PAGE 1

                    SUBROUTINE COEF2
              C
              C  THIS ROUTINE COMPUTES THE INTER-SEAM INFLUENCE COEFFICIENTS. I.E.
              C  THE INFLUENCES OF AN ELEMENT ON OTHER ELEMENTS NOT LYING IN THE
              C  SAME PLANE.
              C
                    COMMON/BK3/    EN(3,3), IMAX, JMAX
```

```
      COMMON/BK4/    S13U(3300), S23U(3300), S33U(3300),
     #               S23V(3300), S33V(3300), S33W(3300),
     #               CF(3,27,4)
      COMMON/BK5/    D1U(3300), D2U(3300), D3U(3300),
     #               D2V(3300), D3V(3300),
     #               D1W(3300), D2W(3300), D3W(3300)
      COMMON/BK7/    XI, ET, ZT, SEG, DERI01, DERI02, DERI03, DERI04,
     #               DERI05, DERI06, DERI07, DERI08, DERI09, DERI10,
     #               DERI11, DERI12, DERI13, DERI14, DERI15, DERI16,
     #               DERI17, DERI18, DERI19
      COMMON/CONSTS/ PI, FACT, V1, V2, CONS, COND, CON, IAMAT(28),
     #               HEW, HBW, EW, RLIM, NBXE, NBT, KB, NTR, ITMINI,
     #               ITMAXI, IPHASE, JOG, ID(4,4), LQR, KI, KE, NCUT,
     #               ACC, NONE, KD0, KD1, KD2, KD3, KD4, KD5, KD6
     #               , NELMX, NELMY, NBLKX, NBLKY, NBLKFM
      COMMON/IN1/    TITLE(20), V, E, NSEAM, NMATS
      COMMON/IN5/    XO(4), YO(4), ZO(4), THIKNS(4)
C
      LAST=NSEAM-1
      NUM = 1600
      KOUNT = 1
C
C  LOOP FOR EACH PAIR OF SEAMS
C
      DO 400 NP=1,LAST
        NEXT=NP+1
C
        DO 300 NS=NEXT,NSEAM
          ZT =   (XO(NP) - XO(NS)) * EN(3,1)
     #         + (YO(NP) - YO(NS)) * EN(3,2)
     #         + (ZO(NP) - ZO(NS)) * EN(3,3)
C
C  FIRST THE BLOCK TO BLOCK COEFFICIENTS
C
          ID(NP,NS) = NUM + 1
```

```
              ID(NS,NP) = ID(NP,NS)
              JAILER = 1
              SEG=HBW
              INDX = 40
   50         CONTINUE
      C
      C..CHECK FOR POSSIBLE MEMORY OVERFLOW
      C
              KOUNT = KOUNT + 1
              IF(KOUNT .LE. 10) GO TO 60
                PRINT 10000
                STOP
      C
   60         DO 200 JP=1,INDX
                XI=(JP-1)*2.*SEG
SUBROUTINE COEF2 74/855 OPT=2 FTN 4.8+587  86/02/05. 13.56.15  PAGE 2

      C
                DO 100 IP=1,INDX
                  ET=(IP-1)*2.*SEG
                  NUM = NUM + 1
                  CALL DERIVE
      C
      C  STRESS COEFFICIENTS--
      C
                  S13U(NUM)=V*DERI04-V1*DERI09+ZT*DERI12
                  S23U(NUM)=V*DERI05+ZT*DERI14
                  S33U(NUM)=ZT*DERI15
                  S23V(NUM)=V*DERI07-V1*DERI09+ZT*DERI17
                  S33V(NUM)=ZT*DERI18
                  S33W(NUM)=ZT*DERI19-DERI09
      C
      C  DISPLACEMENT COEFFICIENTS--
      C
                  D1U(NUM)=2.*V1*DERI03-ZT*DERI04
```

```
                  D2U(NUM)=-ZT*DERIO5
                  D3U(NUM)=V2*DERIO1-ZT*DERIO6
                  D2V(NUM)=2.*V1*DERIO3-ZT*DERIO7
                  D3V(NUM)=V2*DERIO2-ZT*DERIO8
                  D1W(NUM)=-V2*DERIO1-ZT*DERIO6
                  D2W(NUM)=-V2*DERIO2-ZT*DERIO8
                  D3W(NUM)=2.*V1*DERIO3-ZT*DERIO9
  100           CONTINUE
C
  200         CONTINUE
C
C  THEN THE ELEMENT TO ELEMENT COEFFICIENTS--
C  CHECK IF THE SEAMS NP AND NS ARE CLOSE ENOUGH THAT ELEMENT TO
C  ELEMENT INTER SEAM INFLUENCE COEFFICIENTS ARE NEEDED ALSO. IF YES,
C  RETURN TO STMT 50 AND COMPUTE THESE STORE THEM IN THE ELEMENT TO ELEMENT
C  COEFFICIENT LEVELS.
C
              IF (JAILER .NE. 1) GO TO 300
                ZS=ABS(ZT)
                IF (ZS .GT. RLIM) GO TO 300
                  SEG = HEW
                  ID(NP,NS)=-ID(NP,NS)
                  ID(NS,NP)=ID(NP,NS)
                  JAILER = 2
                  INDX = 10
                  GO TO 50
C
  300   CONTINUE
C
  400 CONTINUE
C
10000 FORMAT(//*          TOO MANY SEAMS ARE VERY CLOSE TO EACH OTHER.
     1THE CLOSENESS OR PROXIMITY IS DECIDED BY "RLIM".*/*         TO OBT
     2AIN PROPER RESULT DECREASE THE BLOCK DIMENSIONS OR ARTIFICIALLY SE
     3T RLIM  HIGHER*/*         OR INCREASE THE DIMENSIONS OF THE INFLUE
```

```
                  4NCE COEFFICIENT ARRAYS S11U,...D1V..ETC.-THE*/*        LENGTH OF
                  5 ARRAY "WASTE" IN SUBROUTINE "COEF2" MUST BE READJUSTED FOR */*
                  6       PROPER MATCHING OF MEMORY BLOCKS*)
              C
         SUBROUTINE COEF2 74/855 OPT=2 FTN 4.8+587  86/02/05. 13.56.15  PAGE 3

    115                RETURN
                       END


       SYMBOLIC REFERENCE MAP (R=1)

ENTRY POINTS
    1  COEF2

VARIABLES          TYPE             RELOCATION
  103  ACC         REAL                 CONSTS      46530  CF          REAL        ARRAY   BK4
    6  CON         REAL                 CONSTS          5  COND        REAL                CONSTS
    4  CONS        REAL                 CONSTS          4  DERI01      REAL                BK7
    5  DERI02      REAL                 BK7             6  DERI03      REAL                BK7
    7  DERI04      REAL                 BK7            10  DERI05      REAL                BK7
   11  DERI06      REAL                 BK7            12  DERI07      REAL                BK7
   13  DERI08      REAL                 BK7            14  DERI09      REAL                BK7
   15  DERI10      REAL                 BK7            16  DERI11      REAL                BK7
   17  DERI12      REAL                 BK7            20  DERI13      REAL                BK7
   21  DERI14      REAL                 BK7            22  DERI15      REAL                BK7
   23  DERI16      REAL                 BK7            24  DERI17      REAL                BK7
   25  DERI18      REAL                 BK7            26  DERI19      REAL                BK7
    0  D1U         REAL        ARRAY    BK5         40164  D1W         REAL        ARRAY   BK5
 6344  D2U         REAL        ARRAY    BK5         23254  D2V         REAL        ARRAY   BK5
46530  D2W         REAL        ARRAY    BK5         14710  D3U         REAL        ARRAY   BK5
31620  D3V         REAL        ARRAY    BK5         55074  D3W         REAL        ARRAY   BK5
   25  E           REAL                 IN1             0  EN          REAL        ARRAY   BK3
    1  ET          REAL                 BK7            45  EW          REAL                CONSTS
    1  FACT        REAL                 CONSTS         44  HBW         REAL                CONSTS
```

```
   43  HEW        REAL                 CONSTS           7  IAMAT      INTEGER    ARRAY    CONSTS
   57  ID         INTEGER    ARRAY     CONSTS          11  IMAX       INTEGER             BK3
  265  INDX       INTEGER                             267  IP         INTEGER
   55  IPHASE     INTEGER              CONSTS          54  ITMAXI     INTEGER             CONSTS
   53  ITMINI     INTEGER              CONSTS         264  JAILER     INTEGER
   12  JMAX       INTEGER              BK3             56  JOG        INTEGER             CONSTS
  266  JP         INTEGER                              51  KB         INTEGER             CONSTS
  105  KD0        INTEGER              CONSTS         106  KD1        INTEGER             CONSTS
  107  KD2        INTEGER              CONSTS         110  KD3        INTEGER             CONSTS
  111  KD4        INTEGER              CONSTS         112  KD5        INTEGER             CONSTS
  113  KD6        INTEGER              CONSTS         101  KE         INTEGER             CONSTS
  100  KI         INTEGER              CONSTS         260  KOUNT      INTEGER
  256  LAST       INTEGER                              77  LQR        INTEGER             CONSTS
  120  NBLKFM     INTEGER              CONSTS         116  NBLKX      INTEGER             CONSTS
  117  NBLKY      INTEGER              CONSTS          50  NBT        INTEGER             CONSTS
   47  NBXE       INTEGER              CONSTS         102  NCUT       INTEGER             CONSTS
  114  NELMX      INTEGER              CONSTS         115  NELMY      INTEGER             CONSTS
  262  NEXT       INTEGER                              27  NMATS      INTEGER             IN1
  104  NONE       INTEGER              CONSTS         261  NP         INTEGER
  263  NS         INTEGER                              26  NSEAM      INTEGER             IN1
   52  NTR        INTEGER              CONSTS         257  NUM        INTEGER
    0  PI         REAL                 CONSTS          46  RLIM       REAL                CONSTS
    3  SEG        REAL                 BK7              0  S13U       REAL       ARRAY    BK4
 6344  S23U       REAL       ARRAY     BK4          23254  S23V       REAL       ARRAY    BK4
14710  S33U       REAL       ARRAY     BK4          31620  S33V       REAL       ARRAY    BK4
       SUBROUTINE COEF2 74/855 OPT=2 FTN 4.8+587  86/02/05. 13.56.15  PAGE 4
VARIABLES          TYPE          RELOCATION
4 164  S33W       REAL       ARRAY     BK4             14  THIKNS     REAL       ARRAY    IN5
    0  TITLE      REAL       ARRAY     IN1             24  V          REAL                IN1
    2  V1         REAL                 CONSTS           3  V2         REAL                CONSTS
    0  XI         REAL                 BK7              0  XO         REAL       ARRAY    IN5
    4  YO         REAL       ARRAY     IN5             10  ZO         REAL       ARRAY    IN5
  270  ZS         REAL                                  2  ZT         REAL                BK7

FILE NAMES         MODE
```

```
      OUTPUT    FMT

EXTERNALS           TYPE   ARGS
      DERIVE                 0

INLINE FUNCTIONS    TYPE   ARGS
      ABS        REAL        1  INTRIN

STATEMENT LABELS
   37  50                                   45  60                                      0  100
    0  200                                 147  300                                     0  400
  176  10000   FMT

LOOPS  LABEL    INDEX     FROM-TO     LENGTH     PROPERTIES
   11  400      NP          32 105      152B               EXT REFS  NOT INNER
   15  300      NS          35 103      140B               EXT REFS  NOT INNER
   46  200      JP          56 85        65B               EXT REFS  NOT INNER
   53  100      IP          59 83        55B               EXT REFS
COMMON BLOCKS   LENGTH
      BK3           11
      BK4        20124
      BK5        26400
      BK7           23
      CONSTS        81
      IN1           24
      IN5           16

STATISTICS
  PROGRAM LENGTH                  271B     185
  CM LABELED COMMON LENGTH     133127B   46679
        60000B CM USED
        SUBROUTINE CONSTN 74/855 OPT=2 FTN 4.8+587  86/02/05. 13.56.15  PAGE 1

     1                  SUBROUTINE CONSTN
                 C
```

```
C..THIS SUBROUTINE WILL DEFINE THE PROGRAM CONSTANTS
C  AND OTHER INITIAL PARAMETERS.
C
      COMMON /IN1/ TITLE(20), V, E, NSEAM, NMATS
      COMMON /IN4/ BW, NBXI, NBET, NSYM, NOSP, IFXS, IFXE, IFYS, IFYE
      COMMON/IN4A/   IEX, EXR(10)
      COMMON /IN7/ ORF, ITMAX, NRUN, ITP, NGRID
      COMMON /CONSTS/ PI, FACT, V1, V2, CONS, COND, CON, IAMAT(28),
     #        HEW, HBW, EW, RLIM, NBXE, NBT, KB, NTR, ITMINI,
     #        ITMAXI, IPHASE, JOG, ID(4,4), LQR, KI, KE, NCUT,
     #        ACC, NONE, KD0, KD1, KD2, KD3, KD4, KD5, KD6
     #               , NELMX, NELMY, NBLKX, NBLKY, NBLKFM
C
      DATA ID /-1,  0,  0,  0,
     #          0, -1,  0,  0,
     #          0,  0, -1,  0,
     #          0,  0,  0, -1 /
      DATA IPHASE /6HINPUT /
      DATA JOG    /1/
      DATA LQR    /2026/
      DATA KI /805/
      DATA KE /2100/
      DATA NCUT /20/
      DATA ACC /.001/
      DATA NONE /1/
      DATA KD0 /1/
      DATA KD1 /1/
      DATA KD2 /0/
      DATA KD3 /0/
      DATA KD4 /0/
      DATA KD5 /0/
      DATA KD6 /0/
      DATA PI /3.1415926535897/
C
      FACT   = PI / 180.
```

```
                  V1     = 1. - V
                  V2     = 1. - 2. * V
                  CONS   = E / (8. * PI * (1. - V * V))
                  COND   = 1. / (8. * PI * (1. - V))
                  CON    = -ORF / CONS
                  HEW    = BW / 10.
                  HBW    = BW / 2.
                  EW     = BW / 5.
                  RLIM   = BW
                  NBXE   = NBXI * NBET
                  NBT    = NBXE * NSEAM
                  KB     = ( NBLKX * NBLKY * NSEAM ) * 3
                  NTR    = NBLKX * NBLKY * NSEAM + 4
                  ITMINI = ITP + 1
                  ITMAXI = ITP + ITMAX
                  RETURN
                  END
```

SUBROUTINE CONSTN 74/855 OPT=2 FTN 4.8+587 86/02/05. 13.56.15 PAGE 2

```
    SYMBOLIC REFERENCE MAP (R=1)

ENTRY POINTS
    1  CONSTN

VARIABLES          TYPE          RELOCATION
  103  ACC       REAL                  CONSTS        0  BW       REAL                 IN4
    6  CON       REAL                  CONSTS        5  COND     REAL                 CONSTS
    4  CONS      REAL                  CONSTS       25  E        REAL                 IN1
   45  EW        REAL                  CONSTS        1  EXR      REAL       ARRAY     IN4A
    1  FACT      REAL                  CONSTS       44  HBW      REAL                 CONSTS
   43  HEW       REAL                  CONSTS        7  IAMAT    INTEGER    ARRAY     CONSTS
   57  ID        INTEGER     ARRAY     CONSTS        0  IEX      INTEGER              IN4A
    6  IFXE      INTEGER               IN4           5  IFXS     INTEGER              IN4
   10  IFYE      INTEGER               IN4           7  IFYS     INTEGER              IN4
   55  IPHASE    INTEGER               CONSTS        1  ITMAX    INTEGER              IN7
```

```
  54  ITMAXI    INTEGER             CONSTS      53  ITMINI    INTEGER             CONSTS
   3  ITP       INTEGER             IN7         56  JOG       INTEGER             CONSTS
  51  KB        INTEGER             CONSTS     105  KD0       INTEGER             CONSTS
 106  KD1       INTEGER             CONSTS     107  KD2       INTEGER             CONSTS
 110  KD3       INTEGER             CONSTS     111  KD4       INTEGER             CONSTS
 112  KD5       INTEGER             CONSTS     113  KD6       INTEGER             CONSTS
 101  KE        INTEGER             CONSTS     100  KI        INTEGER             CONSTS
  77  LQR       INTEGER             CONSTS       2  NBET      INTEGER             IN4
 120  NBLKFM    INTEGER             CONSTS     116  NBLKX     INTEGER             CONSTS
 117  NBLKY     INTEGER             CONSTS      50  NBT       INTEGER             CONSTS
  47  NBXE      INTEGER             CONSTS       1  NBXI      INTEGER             IN4
 102  NCUT      INTEGER             CONSTS     114  NELMX     INTEGER             CONSTS
 115  NELMY     INTEGER             CONSTS       4  NGRID     INTEGER             IN7
  27  NMATS     INTEGER             IN1        104  NONE      INTEGER             CONSTS
   4  NOSP      INTEGER             IN4          2  NRUN      INTEGER             IN7
  26  NSEAM     INTEGER             IN1          3  NSYM      INTEGER             IN4
  52  NTR       INTEGER             CONSTS       0  ORF       REAL                IN7
   0  PI        REAL                CONSTS      46  RLIM      REAL                CONSTS
   0  TITLE     REAL      ARRAY     IN1         24  V         REAL                IN1
   2  V1        REAL                CONSTS       3  V2        REAL                CONSTS

COMMON BLOCKS   LENGTH
      IN1           24
      IN4            9
      IN4A          11
      IN7            5
      CONSTS        81

STATISTICS
  PROGRAM LENGTH                  47B      39
  CM LABELED COMMON LENGTH       202B     130
         60000B CM USED
         SUBROUTINE DERIVE 74/855 OPT=2 FTN 4.8+587  86/02/05. 13.56.15  PAGE 1

    1                SUBROUTINE DERIVE
```

```
CC
C  THIS SUBROUTINE OBTAINS THE DERIVATIVES OF THE INTEGRAL FUNCTION I
C
      COMMON/BK7/    XI, ET, ZT, SEG, DERIO1, DERIO2, DERIO3, DERIO4,
     #               DERIO5, DERIO6, DERIO7, DERIO8, DERIO9, DERI10,
     #               DERI11, DERI12, DERI13, DERI14, DERI15, DERI16,
     #               DERI17, DERI18, DERI19
      DATA PI/3.1415926535897/
C
C  DEFINE THE STATEMENT FUNCTIONS
C
      F0(P,R)=ALOG(R-P)
      F1(P,Q,R)=(P*Q)/((R**2-Q**2)*R)
      F2(P,Q,R,T)=Q*T*(P**2*(R**2+P**2)-ZT**2*(R**2-P**2))
      F3(P,Q,R,T)=P*Q*ZT*T*(3.*R**2-Q**2)
C
      XI1=XI-SEG
      XI2=XI+SEG
      ET1=ET-SEG
      ET2=ET+SEG
      R1=SQRT(XI1**2+ET1**2+ZT**2)
      R2=SQRT(XI2**2+ET2**2+ZT**2)
      R3=SQRT(XI1**2+ET2**2+ZT**2)
      R4=SQRT(XI2**2+ET1**2+ZT**2)
      T1=1./((XI1**2+ZT**2)**2*R1**3)
      T2=1./((XI2**2+ZT**2)**2*R2**3)
      T3=1./((XI1**2+ZT**2)**2*R3**3)
      T4=1./((XI2**2+ZT**2)**2*R4**3)
C
C  NOW COMPUTE THE DERIVATIVES
C
      DERIO1=F0(ET2,R3)+F0(ET1,R4)-F0(ET1,R1)-F0(ET2,R2)
      DERIO2=F0(XI1,R3)+F0(XI2,R4)-F0(XI1,R1)-F0(XI2,R2)
C
      IF (ZT .NE. 0) GO TO 100
```

```
            DERIO3=0.
C
            IF(ABS(XI).LE.SEG.AND.ABS(ET).LE.SEG)
     #      DERIO3 = -2 * PI
            GO TO 200
C
  100       DERIO3=ATAN(XI1*ET2/(R3*ZT))+ATAN(XI2*ET1/(R4*ZT))-ATAN(XI1*ET1/
     1         (R1*ZT))-ATAN(XI2*ET2/(R2*ZT))
C
  200 DERIO4=F1(XI1,ET2,R3)+F1(XI2,ET1,R4)-F1(XI1,ET1,R1)-F1(XI2,ET2,R2)
      DERIO5=1./R1+1./R2-1./R3-1./R4
      DERIO6=F1(ZT,ET2,R3)+F1(ZT,ET1,R4)-F1(ZT,ET1,R1)-F1(ZT,ET2,R2)
      DERIO7=F1(ET2,XI1,R3)+F1(ET1,XI2,R4)-F1(ET1,XI1,R1)-F1(ET2,XI2,R2)
      DERIO8=F1(ZT,XI1,R3)+F1(ZT,XI2,R4)-F1(ZT,XI1,R1)-F1(ZT,XI2,R2)
      DERIO9=-DERIO4-DERIO7
      DERI10=F2(XI1,ET1,R1,T1)+F2(XI2,ET2,R2,T2)-F2(XI1,ET2,R3,T3)
     1      -F2(XI2,ET1,R4,T4)
      DERI11=XI1/R3**3+XI2/R4**3-XI1/R1**3-XI2/R2**3
      DERI12=F3(XI1,ET1,R1,T1)+F3(XI2,ET2,R2,T2)-F3(XI1,ET2,R3,T3)
     1      -F3(XI2,ET1,R4,T4)
      DERI13=ET2/R3**3+ET1/R4**3-ET1/R1**3-ET2/R2**3
```

SUBROUTINE DERIVE 74/855 OPT=2 FTN 4.8+587 86/02/05. 13.56.15 PAGE 2

```
      DERI14=ZT*(1./R3**3+1./R4**3-1./R1**3-1./R2**3)
      DERI15=-DERI10-DERI13
      T1=1./((ET1**2+ZT**2)**2*R1**3)
      T2=1./((ET2**2+ZT**2)**2*R2**3)
      T3=1./((ET2**2+ZT**2)**2*R3**3)
      T4=1./((ET1**2+ZT**2)**2*R4**3)
      DERI16=F2(ET1,XI1,R1,T1)+F2(ET2,XI2,R2,T2)-F2(ET2,XI1,R3,T3)
     1      -F2(ET1,XI2,R4,T4)
      DERI17=F3(ET1,XI1,R1,T1)+F3(ET2,XI2,R2,T2)-F3(ET2,XI1,R3,T3)
     1      -F3(ET1,XI2,R4,T4)
      DERI18=-DERI16-DERI11
      DERI19=-DERI12-DERI17
```

```
70          C
                    RETURN
                    END

      SYMBOLIC REFERENCE MAP (R=1)

ENTRY POINTS
    1  DERIVE

VARIABLES       TYPE          RELOCATION
    4  DERI01   REAL              BK7                5  DERI02   REAL              BK7
    6  DERI03   REAL              BK7                7  DERI04   REAL              BK7
   10  DERI05   REAL              BK7               11  DERI06   REAL              BK7
   12  DERI07   REAL              BK7               13  DERI08   REAL              BK7
   14  DERI09   REAL              BK7               15  DERI10   REAL              BK7
   16  DERI11   REAL              BK7               17  DERI12   REAL              BK7
   20  DERI13   REAL              BK7               21  DERI14   REAL              BK7
   22  DERI15   REAL              BK7               23  DERI16   REAL              BK7
   24  DERI17   REAL              BK7               25  DERI18   REAL              BK7
   26  DERI19   REAL              BK7                1  ET       REAL              BK7
  450  ET1      REAL                               451  ET2      REAL
  443  PI       REAL                               452  R1       REAL
  453  R2       REAL                               454  R3       REAL
  455  R4       REAL                                 3  SEG      REAL              BK7
  456  T1       REAL                               457  T2       REAL
  460  T3       REAL                               461  T4       REAL
    0  XI       REAL              BK7              446  XI1      REAL
  447  XI2      REAL                                 2  ZT       REAL              BK7

EXTERNALS        TYPE   ARGS
       ALOG     REAL     1 LIBRARY                      ATAN     REAL        1 LIBRARY
       SQRT     REAL     1 LIBRARY

INLINE FUNCTIONS  TYPE  ARGS
```

```
     ABS        REAL       1  INTRIN                 F0        REAL      2    SF
     F1         REAL       3    SF                   F2        REAL      4    SF
     F3         REAL       4    SF
STATEMENT LABELS
  115  100                             141  200
        SUBROUTINE DERIVE 74/855 OPT=2 FTN 4.8+587  86/02/05. 13.56.15  PAGE 3
COMMON BLOCKS   LENGTH
     BK7            23

STATISTICS
  PROGRAM LENGTH                 462B      306
  CM LABELED COMMON LENGTH        27B       23
        61700B CM USED
        SUBROUTINE ELINFL 74/855 OPT=2 FTN 4.8+587  86/02/05. 13.56.15  PAGE 1

                 SUBROUTINE ELINFL (IP, JP, NP, LPR, MAT, ITER, THICK,
                #                  JB, JE, IB, IE, IND1, IND2, IND3, ERROR
                #                  ,LPE,LPB,LPBFM,IBF,IEF,JBF,JEF)
            C
            C..THIS SUBROUTINE WILL CALCULATE THE INFLUENCE OF ALL ELEMENTS
            C  IN THE BLOCK AND ALL ELEMENTS IN ALL THE BLOCKS IN THE
            C  NEIGHBORHOOD OF THIS BLOCK TO EACH ELEMENT IN THE BLOCK BEING
            C  PROCESSED.  ELEMENT TO ELEMENT COEFFICIENTS ARE USED.
            C
                 COMMON/BK1/    DUB(3200), DVB(3200), DWB(3200),DWBI(3200),
                #               DE(2100),  INDEX(805),  KODE(25), INDX13(805)
                #               ,NGOB(3200)
                 COMMON/BK2/    PO1(4), PO2(4), PO3(4),
                #               P1XI,   P2XI,   P3XI,
                #               P1ET,   P2ET,   P3ET
                 COMMON/BK4/    S13U(3300), S23U(3300), S33U(3300),
                #               S23V(3300), S33V(3300), S33W(3300),
                #               CF(3,27,4)
                 COMMON/BK5/    D1U(3300), D2U(3300), D3U(3300),
                #               D2V(3300), D3V(3300),
```

```
     #                  D1W(3300), D2W(3300), D3W(3300)
      COMMON/CONSTS/ PI, FACT, V1, V2, CONS, COND, CON, IAMAT(28),
     #                  HEW, HBW, EW, RLIM, NBXE, NBT, KB, NTR, ITMINI,
     #                  ITMAXI, IPHASE, JOG, ID(4,4), LQR, KI, KE, NCUT,
     #                  ACC, NONE, KD0, KD1, KD2, KD3, KD4, KD5, KD6
     #                  , NELMX, NELMY, NBLKX, NBLKY, NBLKFM
      COMMON/IN1/       TITLE(20), V, E, NSEAM, NMATS
      COMMON/IN2/       ES(28), GS(28), COHES(28),
     #                  PHI(28), COHES1(28), PHI1(28), IGOB(28)
      COMMON/IN4/       BW, NBXI, NBET, NSYM, NOSP, IFXS, IFXE, IFYS, IFYE
      COMMON/IN4A/      IEX, EXR(10)
      COMMON/AK1/       S13PB, S23PB, S33PB, UPB, VPB, WPB,
     #                  S13PE, S23PE, S33PE, UPE, VPE, WPE
      COMMON/AK2/       SIG1(25), SIG2(25), SIG3(25),
     #                  UPOS(25), VPOS(25), WPOS(25),
     #                  UNEG(25), VNEG(25), WNEG(25)
      DIMENSION USBE(12), MAT(25), MAT2(25), MAT3(25)
      EQUIVALENCE(S13PB, USBE(1))
C
      LOGICAL CORSBL
C
      LPE = 0
      DO 5000 LP=1,5
        DO 4000 KP = 1,5
          LPE = LPE + 1
          KOD = KODE(LPE) * KD0 + KD6
          IF (KOD .EQ. 2) GO TO 4000
          KOD = KODE(LPE)
          DO 50 JT = 7,12
            USBE(JT) = 0.0
   50     CONTINUE
C
          LPEU = LPR + (LP-1) * 5 + KP -1
          LPEV = LPEU + 25
          LPEW = LPEV + 25
```

```
              JLP = (JP - 1) * 5 + LP - 1
              IKP = (IP - 1) * 5 + KP - 1
```



```
C
C..PRIMITIVE STRESS COMPONENTS
C
      P1 = PO1(NP) + JLP * P1XI + IKP * P1ET
      P2 = PO2(NP) + JLP * P2XI + IKP * P2ET
      P3 = PO3(NP) + JLP * P3XI + IKP * P3ET
C
      IF (KOD .LE. 1) GO TO 110
        IMAT = MAT(LPE)
        IF (IMAT .LE. NMATS) GO TO 70
          IF (IMAT .LT. 27) ICODE = IMAT + 2
          IF (IMAT .GT. 26) ICODE = IMAT - 26
          PRINT 10000, ITER,NP,JP,IP,LP,KP,KOD,IAMAT(ICODE)
          STOP
   70   C = COHES(IMAT)
        C1 = COHES1(IMAT)
        TPHI = TAN(PHI(IMAT) * FACT)
        TPHI1 = TAN(PHI1(IMAT) * FACT)
        STFS = -GS(IMAT) / THICK
        STFN = -ES(IMAT) / THICK
        PRIMC = P3 / STFN
        PC = P3 / ES(IMAT)
C
C..ELEMENT INITIAL CLOSURE
C
        EI = PC * THICK * IGOB(IMAT)
C
C
C..NEXT CONSIDER THE INFLUENCE OF ALL ELEMENTS IN THE NEIGHBORHOOD
C
C  OF THE BLOCK(IP,JP,NP)
```

```
      C
        110     KIN = 0
                LPBI = (NP-1) * NBXE + (JP-1) * NBET + IP
                DO 600 NS = 1,NSEAM
                  IF (ID(NP,NS) .GE. 0) GO TO 600
                  IF (ID(NP,NS) .NE. -1) GO TO 120
                    NDEL = 0
                    FACTR = 5.0
                    NROWS = 40
                  GO TO 140
        120         NDEL = 1600
                    FACTR = 1.0
                    NROWS = 10
        140       NUMB = IABS(ID(NP,NS)) + NDEL
                  NPS = NS - NP
                  PZT = FLOAT(ISIGN(NONE,NPS))
      C
      C..SET FLAGS FOR SAME OR DIFFERENT SEAMS
      C
                  ISAME = 0
                  IDIFF = 0
                  ILOWER = 0
                  IF(NS .LT. NP) ILOWER = 1
                  IF(NS .EQ. NP) ISAME = 1
                  IF(NS .NE. NP) IDIFF = 1
      C
SUBROUTINE ELINFL 74/855 OPT=2 FTN 4.8+587  86/02/05. 13.56.15  PAGE 3

                  LSBO = (NS-1) * NBXE
                    LSBOFM = (NS-1) * NBLKFM
      C
      C..CALCULATE INDEX TO THE BLOCK ABOVE OR BELOW THE CURRENT BLOCK
      C
                  LXB = LSBO + (JP-1) * NBET + IP
                  LXBREC = LSBOFM + (JP - IFXS) * NBLKY + IP - IFYS + 1
```

```
C
                GC  = 0
                IF(NS .NE. NP .AND. NGOB(LXB) .GT. 0)
      #           GC = 25 * DWBI(LXB) / NGOB(LXB)
C
                CALL READMS(13,MAT2,25,LXBREC)
                IMAT2 = MAT2(5*LP+KP-5)
                EC    = GC * IGOB(IMAT2)
                DO 500 JS = JB,JE
                  DO 400 IS = IB,IE
                    LSB = LSBO + (JS-1) * NBET + IS
                    GC3 = 0
                    IF(NS.NE.NP.AND.NGOB(LSB).GT.0)
      #               GC3 = 25. * DWBI(LSB) / NGOB(LSB)
C
C..INITIALIZE LOOP VARIABLES
C
                    DEI = DWBI(LPBI)
                    DEC = DWBI(LXB)
                    DEU = DUB(LSB)
                    DEV = DVB(LSB)
                    DEW = DWB(LSB)
                    EC3 = GC3
                    CORSBL = .TRUE.
C
C
C..CHECK TO SEE IF THE INFLUENCING BLOCK IS A FINE MESH BLOCK
C  IF IT IS THEN TURN OFF THE COARSE BLOCK FLAG
C
                    IF (IS .GE. IBF .AND. IS .LE. IEF .AND.
      #                 JS .GE. JBF .AND. JS .LE. JEF) CORSBL=.FALSE.
C
C..ROUTINE TO CALCULATE INFLUENCE OF A NEIGHBORHOOD BLOCK TO THE
C  ELEMENT BEING PROCESSED
C
```

```
                IF (CORSBL) GO TO 142
                LSBREC = LSBOFM + (JS-IFXS) * NBLKY + IS - IFYS + 1
                CALL READMS(13,MAT3,25,LSBREC)
                KIN = KIN + 1
                LSR = (KIN-1) * 75 + 1
                DEI = EI
                DEC = EC
      C
        142     DO 300 LS = 1,5
                  LPS = 5 * (JP - JS) + LP -LS
                  PXI = FLOAT(ISIGN(NONE,LPS))
                  LPS = IABS(LPS)
                  DO 200 KS = 1,5
                    KPS = 5 * (IP - IS) + KP - KS
                    PET = FLOAT(ISIGN(NONE,KPS))
```

SUBROUTINE ELINFL 74/855 OPT=2 FTN 4.8+587 86/02/05. 13.56.15 PAGE 4

```
                    PXE  = PXI * PET
                    PXZ  = PXI * PZT
                    PEZ  = PET * PZT
                    PXEZ  = PXE * PZT
                    KPS  = IABS(KPS)
                    NUM   = NUMB + LPS * NROWS + KPS
                   IF (CORSBL) GO TO 144
                    IMAT3  = MAT3(5*LS+KS-5)
                    EC3    = GC3 * IGOB(IMAT3)
                    LSEU = LSR + (LS-1) * 5 + KS - 1
                    LSEV = LSEU + 25
                    LSEW = LSEV + 25
                    DEU = DE(LSEU)
                    DEV = DE(LSEV)
                    DEW = DE(LSEW)
      C
      C..STRESS CONPONENTS
      C
```

```
  144            S13PE = S13PE + (S13U(NUM) * DEU
     #                         +  S23U(NUM) * DEV *PXE
     #                         +  S33U(NUM) * PXZ *
     #                            (DEW-DEI*ISAME-DEC*IDIFF)
     #                           ) * FACTR
                 S23PE = S23PE + (S23U(NUM) * DEU *PXE
     #                         +  S23V(NUM) * DEV
     #                         +  S33V(NUM) * PEZ *
     #                            (DEW-DEI*ISAME-DEC*IDIFF)
     #                           ) * FACTR
                 S33PE = S33PE + (S33U(NUM) * DEU *PXZ
     #                         +  S33V(NUM) * DEV *PEZ
     #                         +  S33W(NUM) *
     #                            (DEW-DEI*ISAME-DEC*IDIFF)
     #                           ) * FACTR
C
C..DISPLACEMENT COORDINATES
C
                  IF (IPHASE .EQ. 6HINPUT ) GO TO 200
C
C..DEFINE REVISED ELEMENT CLOSURE
C
                   IF(IDIFF .EQ. 1) GO TO 150
C
C..INFLUENCE OF A GOB ELEMENT ON ITSELF
C
                   RC = DEW - DEI
                   IF(JP .EQ. JS .AND. IP .EQ. IS .AND.
     #                LP .EQ. LS .AND. KP .EQ. KS .AND.
     #                IGOB(IMAT) .EQ. 1 .AND. RC .LT. 0)
     #                RC = 0
                      GO TO 190
  150                 RC = DEW - EC3
                      IF(EC3 .EQ. 0) GO TO 190
                      IF(ILOWER .EQ. 1) GO TO 170
```

```
      C
      C..INFLUENCE OF GOB ELEMENT IN A HIGHER SEAM
      C
                            IF(RC .LT. 0) RC = 0
SUBROUTINE ELINFL 74/855 OPT=2 FTN 4.8+587  86/02/05. 13.56.15  PAGE 5

                            GO TO 190
      C
      C..INFLUENCE OF GOB ELEMENT IN A LOWER SEAM
      C
        170                 IF(DEW.LT.EC3) RC=2*DEW
      C
      C..IF DEW > EC3 THEN RC = 2*EC3 + (DEW - EC3)
      C  THERETORE
                            IF(DEW.GE.EC3) RC=DEW + EC3
        190                 CONTINUE
                         UPE = UPE + D1U(NUM) * DEU * PZT
          #                        + D2U(NUM) * DEV * PXEZ
          #                        + D1W(NUM) * PXI * DEW
                         VPE = VPE + D2U(NUM) * DEU * PXEZ
          #                        + D2V(NUM) * DEV * PZT
          #                        + D2W(NUM) * PET * DEW
                        TEMP =  D3U(NUM) * DEU * PXI
          #                        + D3V(NUM) * DEV * PET
          #                        + D3W(NUM) * PZT * RC
                        IF(IDIFF .EQ. 1) TEMP = -TEMP
                        WPE = WPE + TEMP
        200           CONTINUE
        300         CONTINUE
        400       CONTINUE
        500     CONTINUE
        600   CONTINUE
      C
      C..NOW OBTAIN THE TOTAL INFLUENCE AT THIS ELEMENT (INDUCED STRESSES)
      C
```

```
              T1 = (S13PB + S13PE) * CONS
              T2 = (S23PB + S23PE) * CONS
              T3 = (S33PB + S33PE) * CONS
      C
      C..OBTAIN THE STRESS RESIDUES AT THIS ELEMENT
      C
              GO TO (1000,4000,1100,1200,1200,1200,1200,1200), KOD
      C
      C..MINED OUT ELEMENT
      C
       1000   S13 = P1 + T1
              S23 = P2 + T2
              S33 = P3 + T3
              GOTO 2100
      C
      C..UNMINED AND DEFORMABLE ELEMENT
      C
       1100   S13 = STFS * DE(LPEU) + T1
              S23 = STFS * DE(LPEV) + T2
              S33 = STFN * DE(LPEW) + T3 + ( P3 * IGOB(IMAT) )
      C
              GO TO 2200
      C
      C..MOHR COULOMB ELEMENT
      C
       1200   MODE = KOD - 3
              GO TO (1300,1300,1400,1500,1500), MODE
      C
SUBROUTINE ELINFL 74/855 OPT=2 FTN 4.8+587  86/02/05. 13.56.15  PAGE 6

      C..UNCRACKED, UNYIELDED, OR YIELDED
      C
       1300   SYIELD = C + (P3 + T3) * TPHI
              KD = KD2
              GO TO 1600
```

```
C
C..CRACKED, OPEN, CHECK IF CLOSED BACK AGAIN, IF NOT, TREAT AS MINED OUT
C
 1400     IF (DE(LPEW) .LT. PRIMC) GO TO 1700
          KODE(LPE) = KODE(LPE) * KD1 + KD4
C
C..CRACKED OPEN AND CLOSED BACK AGAIN. YIELDED AND UNYIELDED
C
 1500     SYIELD = C1 + (P3 + T3) * TPHI1
          MODE = MODE - 3
          KD = KD5
C
C..CHECK IF CRACKING CONDITIONS EXIST. IF SO, TREAT AS MINED OUT
C
 1600     IF (SYIELD .GE. 0) GO TO 1800
 1700     S13 = P1 + T1
          S23 = P2 + T2
          S33 = P3 + T3
          KODE(LPE) = KODE(LPE) * KD1 + KD3
          GO TO 2100
C
C..CHECK IF YIELD CONDITION EXISTS. IF NO, TREAT AS UNMINED DEFORMABLE
C
 1800     STOTAL = SQRT((T1 + P1) ** 2 + (P2 + T2) ** 2)
          SDIFF = STOTAL - SYIELD
          IF (STOTAL .EQ. 0.0) GO TO 1900
          SRATIO = SDIFF / STOTAL
          IF (SDIFF .GE. 0 .OR. MODE .EQ. 2) GO TO 2000
 1900     S13 = STFS * DE(LPEU) + T1
          S23 = STFS * DE(LPEV) + T2
          S33 = STFN * DE(LPEW) + T3 + ( P3 * IGOB(IMAT) )
          GO TO 2200
C
C..YIELD CONDITION EXISTS. S13,S23 CORRESPOND TO
C  OVERSHOOT FROM MOHR-COULOMB ENVELOPE
```

```
C
 2000     S13 = SRATIO * (P1 + T1)
          S23 = SRATIO * (P2 + T2)
          S33 = STFN * DE(LPEW) + T3 + ( P3 * IGOB(IMAT) )
          KODE(LPE) = KODE(LPE) * KD1 + KD
          GO TO 2100
C
C..OBTAIN THE RIDE AND CLOSURE CORRECTIONS
C  CHECK FOR THE COMPLETE CLOSURE AT THIS ELEMENT
C  IF YES, SET THE CLOSURE CORRECTION ACCORDINGLY
C
 2100     DELU = S13 * CF(1,27,NP)
          DELV = S23 * CF(2,27,NP)
          DELW = S33 * CF(3,27,NP)
          IF (DE(LPEW) .GE. THICK) DELW = THICK - DE(LPEW)
          GO TO 2300
 2200     DELU = S13 * CF(1,IMAT,NP)
```


```
          DELV = S23 * CF(2,IMAT,NP)
          DELW = S33 * CF(3,IMAT,NP)
C
C
C..OBTAIN THE ERRORS IN THE RIDE AND CLOSURE VALUES
C
 2300     EROR1 = ABS(DELU)
          EROR2 = ABS(DELV)
          EROR3 = ABS(DELW)
          IF (EROR1 .GT. ACC) IND1 = IND1 + 1
          IF (EROR2 .GT. ACC) IND2 = IND2 + 1
          IF (EROR3 .GT. ACC) IND3 = IND3 + 1
          ERROR = AMAX1(ERROR, EROR1, EROR2, EROR3)
C
C..IF THIS IS THE OUTPUT CYCLE, OBTAIN THE TOTAL STRESSES AND
C  DISPLACEMENTS AT THIS ELEMENT
```

```
C
C
          IF (IPHASE .EQ. 6HINPUT ) GO TO 2400
            SIG1(LPE) = P1 + T1
            SIG2(LPE) = P2 + T2
            SIG3(LPE) = P3 + T3
            UPOS(LPE) = (UPB + UPE) * COND
            VPOS(LPE) =  (VPB + VPE) * COND
            WNEG(LPE) = -1 * (WPB + WPE) * COND
            UNEG(LPE) = UPOS(LPE) + DE(LPEU)
            VNEG(LPE) = VPOS(LPE) + DE(LPEV)
            WPOS(LPE) = WNEG(LPE) - DE(LPEW)
          GO TO 4000
C
C..OBTAIN THE CORRECTED VALUES OF RIDES AND CLOSURES
C
 2400       DE(LPEU) = DE(LPEU) + DELU
            DE(LPEV) = DE(LPEV) + DELV
            DE(LPEW) = DE(LPEW) + DELW
            DUB(LPB) = DUB(LPB) + DELU / 25.
            DVB(LPB) = DVB(LPB) + DELV / 25.
            DWB(LPB) = DWB(LPB) + DELW / 25.
 4000   CONTINUE
 5000 CONTINUE
C
C
C..FORMATS
C
10000 FORMAT(//* --- ERROR --- PROGRAM STOPS ---*//
     C* UNDEFINED OR UNACCEPTABLE  PROPERTY SET FOR THIS ELEMENT*//
     C* ITERATION: *,I5/
     C* SEAM NO. : *,I5/
     C* BLOCK JP,IP:*,2I5/
     C* ELEMENT LP,KP:*,2I5/
     C* ELEMENT KODE: *,I5/
```

```
            C* ELEMENT PROPERTY: *,A1//
395         C* --- BETTER SUCCESS NEXT TIME ---*)
      C
             RETURN
             END
```

```
      SUBROUTINE ELINFL 74/855 OPT=2 FTN 4.8+587  86/02/05. 13.56.15  PAGE 8
      SYMBOLIC REFERENCE MAP (R=1)
ENTRY POINTS
    3  ELINFL
VARIABLES          TYPE           RELOCATION
  103  ACC       REAL                CONSTS        0  BW        REAL                IN4
 1204  C         REAL                          46530  CF        REAL      ARRAY     BK4
   70  COHES     REAL      ARRAY     IN2         160  COHES1    REAL      ARRAY     IN2
    6  CON       REAL                CONSTS        5  COND      REAL                CONSTS
    4  CONS      REAL                CONSTS     1165  CORSBL    LOGICAL
 1205  C1        REAL                          31000  DE        REAL      ARRAY     BK1
 1245  DEC       REAL                           1244  DEI       REAL
 1311  DELU      REAL                           1312  DELV      REAL
 1313  DELW      REAL                           1246  DEU       REAL
 1247  DEV       REAL                           1250  DEW       REAL
    0  DUB       REAL      ARRAY     BK1        6200  DVB       REAL      ARRAY     BK1
14400  DWB       REAL      ARRAY     BK1       22600  DWBI      REAL      ARRAY     BK1
    0  D1U       REAL      ARRAY     BK5       40164  D1W       REAL      ARRAY     BK5
 6344  D2U       REAL      ARRAY     BK5       23254  D2V       REAL      ARRAY     BK5
46530  D2W       REAL      ARRAY     BK5       14710  D3U       REAL      ARRAY     BK5
31620  D3V       REAL      ARRAY     BK5       55074  D3W       REAL      ARRAY     BK5
   25  E         REAL                IN1        1237  EC        REAL
 1251  EC3       REAL                           1214  EI        REAL
 1314  EROR1     REAL                           1315  EROR2     REAL
 1316  EROR3     REAL                              0  ERROR     REAL                  F.P.
    0  ES        REAL      ARRAY     IN2          45  EW        REAL                CONSTS
    1  EXR       REAL      ARRAY     IN4A          1  FACT      REAL                CONSTS
 1221  FACTR     REAL                           1235  GC        REAL
 1243  GC3       REAL                             34  GS        REAL      ARRAY     IN2
   44  HBW       REAL                CONSTS       43  HEW       REAL                CONSTS
    7  IAMAT     INTEGER   ARRAY     CONSTS        0  IB        INTEGER               F.P.
    0  IBF       INTEGER               F.P.     1203  ICODE     INTEGER
   57  ID        INTEGER   ARRAY     CONSTS     1227  IDIFF     INTEGER
    0  IE        INTEGER               F.P.        0  IEF       INTEGER               F.P.
    0  IEX       INTEGER             IN4A          6  IFXE      INTEGER             IN4
    5  IFXS      INTEGER             IN4          10  IFYE      INTEGER             IN4
```

```
      7  IFYS      INTEGER                 IN4          250  IGOB      INTEGER    ARRAY      IN2
   1176  IKP       INTEGER                             1230  ILOWER    INTEGER
   1202  IMAT      INTEGER                             1236  IMAT2     INTEGER
   1267  IMAT3     INTEGER                            35064  INDEX     INTEGER    ARRAY      BK1
  36562  INDX13    INTEGER    ARRAY        BK1            0  IND1      INTEGER                F.P.
      0  IND2      INTEGER                  F.P.          0  IND3      INTEGER                F.P.
      0  IP        INTEGER                  F.P.         55  IPHASE    INTEGER               CONSTS
   1241  IS        INTEGER                             1226  ISAME     INTEGER
      0  ITER      INTEGER                  F.P.         54  ITMAXI    INTEGER               CONSTS
     53  ITMINI    INTEGER                 CONSTS         0  JB        INTEGER                F.P.
      0  JBF       INTEGER                  F.P.          0  JE        INTEGER                F.P.
      0  JEF       INTEGER                  F.P.       1175  JLP       INTEGER
     56  JOG       INTEGER                 CONSTS         0  JP        INTEGER                F.P.
   1240  JS        INTEGER                             1171  JT        INTEGER
     51  KB        INTEGER                 CONSTS      1305  KD        INTEGER
    105  KD0       INTEGER                 CONSTS       106  KD1       INTEGER               CONSTS
    107  KD2       INTEGER                 CONSTS       110  KD3       INTEGER               CONSTS
    111  KD4       INTEGER                 CONSTS       112  KD5       INTEGER               CONSTS
    113  KD6       INTEGER                 CONSTS       101  KE        INTEGER               CONSTS
         SUBROUTINE ELINFL 74/855 OPT=2 FTN 4.8+587  86/02/05. 13.56.15  PAGE 9
VARIABLES             TYPE              RELOCATION
    100  KI        INTEGER                 CONSTS      1215  KIN       INTEGER
   1170  KOD       INTEGER                            36531  KODE      INTEGER    ARRAY      BK1
   1167  KP        INTEGER                             1260  KPS       INTEGER
   1257  KS        INTEGER                             1166  LP        INTEGER
      0  LPB       INTEGER                  F.P.          0  LPBFM     INTEGER   *UNUSED      F.P.
   1216  LPBI      INTEGER                                0  LPE       INTEGER                F.P.
   1172  LPEU      INTEGER                             1173  LPEV      INTEGER
   1174  LPEW      INTEGER                                0  LPR       INTEGER                F.P.
   1255  LPS       INTEGER                               77  LQR       INTEGER               CONSTS
   1254  LS        INTEGER                             1242  LSB       INTEGER
   1252  LSBREC    INTEGER                             1231  LSBO      INTEGER
   1232  LSBOFM    INTEGER                             1270  LSEU      INTEGER
   1271  LSEV      INTEGER                             1272  LSEW      INTEGER
   1253  LSR       INTEGER                             1233  LXB       INTEGER
   1234  LXBREC    INTEGER                                0  MAT       INTEGER    ARRAY       F.P.
```

1317	MAT2	INTEGER	ARRAY		1350	MAT3	INTEGER	ARRAY	
1303	MODE	INTEGER			2	NBET	INTEGER		IN4
120	NBLKFM	INTEGER		CONSTS	116	NBLKX	INTEGER		CONSTS
117	NBLKY	INTEGER		CONSTS	50	NBT	INTEGER		CONSTS
47	NBXE	INTEGER		CONSTS	1	NBXI	INTEGER		IN4
102	NCUT	INTEGER		CONSTS	1220	NDEL	INTEGER		
114	NELMX	INTEGER		CONSTS	115	NELMY	INTEGER		CONSTS
40227	NGOB	INTEGER	ARRAY	BK1	27	NMATS	INTEGER		IN1
104	NONE	INTEGER		CONSTS	4	NOSP	INTEGER		IN4
0	NP	INTEGER		F.P.	1224	NPS	INTEGER		
1222	NROWS	INTEGER			1217	NS	INTEGER		
26	NSEAM	INTEGER		IN1	3	NSYM	INTEGER		IN4
52	NTR	INTEGER		CONSTS	1266	NUM	INTEGER		
1223	NUMB	INTEGER			1213	PC	REAL		
1261	PET	REAL			1264	PEZ	REAL		
124	PHI	REAL	ARRAY	IN2	214	PHI1	REAL	ARRAY	IN2
0	PI	REAL		CONSTS	0	PO1	REAL	ARRAY	BK2
4	PO2	REAL	ARRAY	BK2	10	PO3	REAL	ARRAY	BK2
1212	PRIMC	REAL			1262	PXE	REAL		
1265	PXEZ	REAL			1256	PXI	REAL		
1263	PXZ	REAL			1225	PZT	REAL		
1177	P1	REAL			17	P1ET	REAL		BK2
14	P1XI	REAL		BK2	1200	P2	REAL		
20	P2ET	REAL		BK2	15	P2XI	REAL		BK2
1201	P3	REAL			21	P3ET	REAL		BK2
16	P3XI	REAL		BK2	1273	RC	REAL		
46	RLIM	REAL		CONSTS	1307	SDIFF	REAL		
0	SIG1	REAL	ARRAY	AK2	31	SIG2	REAL	ARRAY	AK2
62	SIG3	REAL	ARRAY	AK2	1310	SRATIO	REAL		
1211	STFN	REAL			1210	STFS	REAL		
1306	STOTAL	REAL			1304	SYIELD	REAL		
1300	S13	REAL			0	S13PB	REAL		AK1
6	S13PE	REAL		AK1	0	S13U	REAL	ARRAY	BK4
1301	S23	REAL			1	S23PB	REAL		AK1
7	S23PE	REAL		AK1	6344	S23U	REAL	ARRAY	BK4
23254	S23V	REAL	ARRAY	BK4	1302	S33	REAL		

```
    2  S33PB    REAL               AK1         10  S33PE    REAL              AK1
14710  S33U     REAL      ARRAY    BK4      31620  S33V     REAL     ARRAY    BK4
40164  S33W     REAL      ARRAY    BK4       1274  TEMP     REAL
    0  THICK    REAL                F.P.        0  TITLE    REAL     ARRAY    IN1
 1206  TPHI     REAL                         1207  TPHI1    REAL
 1275  T1       REAL                         1276  T2       REAL
       SUBROUTINE ELINFL 74/855 OPT=2 FTN 4.8+587  86/02/05. 13.56.15  PAGE 10
VARIABLES        TYPE          RELOCATION
 1277  T3       REAL                          226  UNEG     REAL     ARRAY    AK2
    3  UPB      REAL               AK1         11  UPE      REAL              AK1
  113  UPOS     REAL      ARRAY    AK2          0  USBE     REAL     ARRAY    AK1
   24  V        REAL               IN1        257  VNEG     REAL     ARRAY    AK2
    4  VPB      REAL               AK1         12  VPE      REAL              AK1
  144  VPOS     REAL      ARRAY    AK2          2  V1       REAL              CONSTS
    3  V2       REAL               CONSTS     310  WNEG     REAL     ARRAY    AK2
    5  WPB      REAL               AK1         13  WPE      REAL              AK1
  175  WPOS     REAL      ARRAY    AK2
FILE NAMES       MODE
       OUTPUT   FMT
EXTERNALS         TYPE   ARGS
       READMS             4                         SQRT     REAL     1 LIBRARY
       TAN      REAL      1 LIBRARY
INLINE FUNCTIONS  TYPE   ARGS
       ABS      REAL      1  INTRIN                 AMAX1    REAL     0  INTRIN
       FLOAT    REAL      1  INTRIN                 IABS     INTEGER  1  INTRIN
       ISIGN    INTEGER   2  INTRIN
STATEMENT LABELS
    0  50                        103  70                                124  110
  144  120                       150  140                               305  142
  372  144                       436  150                               444  170
  452  190                       505  200                                 0  300
    0  400                         0  500                               521  600
  550  1000                      557  1100                              574  1200
  607  1300                      615  1400                              623  1500
  632  1600                      634  1700                              646  1800
  661  1900                      676  2000                              714  2100
```

```
 730  2200                              741  2300                          1021  2400
1037  4000                                0  5000                          1121  10000   FMT
LOOPS  LABEL    INDEX     FROM-TO    LENGTH    PROPERTIES
  13   5000     LP          43 382    1033B              EXT REFS  NOT INNER
  20   4000     KP          44 381    1023B              EXT REFS  NOT INNER
  30   50       JT          49 51        2B        OPT
 134   600      NS          92 254     370B              EXT REFS  NOT INNER
 230   500      JS         130 253     271B              EXT REFS  NOT INNER
 233   400      IS         131 252     263B              EXT REFS  NOT INNER
 331   300      LS         165 251     162B              NOT INNER
 352   200      KS         169 250     135B        OPT
COMMON BLOCKS   LENGTH
       BK1       19735
       BK2          18
       BK4       20124
       BK5       26400
       CONSTS       81
       IN1          24
       IN2         196
       IN4           9
       IN4A         11
       AK1          12
       AK2         225
       SUBROUTINE ELINFL 74/855 OPT=2 FTN 4.8+587  86/02/05. 13.56.15  PAGE 11
STATISTICS
  PROGRAM LENGTH                 1455B     813
  CM LABELED COMMON LENGTH     202423B   66835
        61100B CM USED
       SUBROUTINE INFCOF 74/855 OPT=2 FTN 4.8+587  86/02/05. 13.56.15  PAGE 1
                      SUBROUTINE INFCOF
                C
                C..THIS SUBROUTINE WILL EITHER READ IN THE INFLUENCE
                C  COEFFICIENTS CALCULATED AND STORED IN A PRIOR RUN,
                C  OR CALCULATE THE COEFFICIENTS IF THIS IS THE MAIDEN RUN.
                C
                C  THE COEFFICIENTS CALCULATED IN THE MAIDEN RUN WILL BE WRITTEN
```

```
C  TO DISK FOR ANY FUTURE RUNS ON THE SAME GRID.
C
      COMMON/BK4/    S13U(3300), S23U(3300), S33U(3300),
     #               S23V(3300), S33V(3300), S33W(3300),
     #               CF(3,27,4)
      COMMON/BK5/    D1U(3300), D2U(3300), D3U(3300),
     #               D2V(3300), D3V(3300),
     #               D1W(3300), D2W(3300), D3W(3300)
      COMMON/CONSTS/ PI, FACT, V1, V2, CONS, COND, CON, IAMAT(28),
     #               HEW, HBW, EW, RLIM, NBXE, NBT, KB, NTR, ITMINI,
     #               ITMAXI, IPHASE, JOG, ID(4,4), LQR, KI, KE, NCUT,
     #               ACC, NONE, KD0, KD1, KD2, KD3, KD4, KD5, KD6
     #               , NELMX, NELMY, NBLKX, NBLKY, NBLKFM
      COMMON/IN1/    TITLE(20), V, E, NSEAM, NMATS
      COMMON/IN2/    ES(28), GS(28), COHES(28),
     #               PHI(28), COHES1(28), PHI1(28), IGOB(28)
      COMMON/IN5/    XO(4), YO(4), ZO(4), THIKNS(4)
      COMMON/IN7/    ORF, ITMAX, NRUN, ITP, NGRID
C
C  CHECK IF OLD OR NEW GRID--FOR NEW: NGRID.EQ.0
C                            FOR OLD: NGRID.NE.0
      IF (NGRID .NE. 0) GO TO 200
C
C  THIS IS A NEW GRID. COMPUTE AFRESH, STORE THE COEFFICIENTS ON FILES
C  TAPE16=COFFS AND TAPE20=COFFSD.
C
C  FIRST THE INTRA-SEAM COEFFICIENTS.
C  ONLY BLOCK TO BLOCK (B-B) COEFFICIENTS ARE COMPUTED. ELEMENT TO
C  ELEMENT COEFFICIENTS ARE CONSTANT FACTORS OF B-B COEFFICIENTS.
C
         CALL COEF1
C
C  THEN THE INTER-SEAM COEFFICIENTS.
C
         IF ( NSEAM .GT. 1) CALL COEF2
C
```

```
C  THEN THE RIDE AND CLOSURE CORRECTION FACTORS
C
         SS1=S13U(1)*5.
         SS2=S23V(1)*5.
         SS3=S33W(1)*5.
C
C  FOR MINED ELEMENTS--
C
C
         DO 140 NP=1,NSEAM
           CF(1,27,NP)=CON/SS1
           CF(2,27,NP)=CON/SS2
           CF(3,27,NP)=CON/SS3
  140    CONTINUE
```

SUBROUTINE INFCOF 74/855 OPT=2 FTN 4.8+587 86/02/05. 13.56.15 PAGE 2

```
C
C  FOR UNMINED DEFORMABLE ELEMENTS--
C
         DO 150 NP=1,NSEAM
           DO 150 NPM=1,NMATS
             CF(1,NPM,NP) = CON/(SS1 + (-GS(NPM) / THIKNS(NP) / CONS ))
             CF(2,NPM,NP) = CON/(SS2 + (-GS(NPM) / THIKNS(NP) / CONS ))
             CF(3,NPM,NP) = CON/(SS3 + (-ES(NPM) / THIKNS(NP) / CONS ))
  150    CONTINUE
C
C
         WRITE (16) S13U,S23U,S33U,S23V,S33V,S33W,CF,ID
         WRITE (20) D1U,D2U,D3U,D2V,D3V,D1W,D2W,D3W
       GO TO 999
C
C  THIS IS AN OLD GRID. RETRIEVE THE COEFFICIENTS FROM FILES
C  TAPE16=COFFS AND TAPE20=COFFSD.
C
  200    READ (16) S13U,S23U,S33U,S23V,S33V,S33W,CF,ID
         READ (20) D1U,D2U,D3U,D2V,D3V,D1W,D2W,D3W
C
```

```
                  999 RETURN
     80               END

        SYMBOLIC REFERENCE MAP (R=1)
ENTRY POINTS
     1  INFCOF
VARIABLES        TYPE          RELOCATION
   103  ACC        REAL                 CONSTS        46530  CF        REAL        ARRAY    BK4
    70  COHES      REAL        ARRAY    IN2             160  COHES1    REAL        ARRAY    IN2
     6  CON        REAL                 CONSTS            5  COND      REAL                 CONSTS
     4  CONS       REAL                 CONSTS            0  D1U       REAL        ARRAY    BK5
 40164  D1W        REAL        ARRAY    BK5            6344  D2U       REAL        ARRAY    BK5
 23254  D2V        REAL        ARRAY    BK5           46530  D2W       REAL        ARRAY    BK5
 14710  D3U        REAL        ARRAY    BK5           31620  D3V       REAL        ARRAY    BK5
 55074  D3W        REAL        ARRAY    BK5              25  E         REAL                 IN1
     0  ES         REAL        ARRAY    IN2              45  EW        REAL                 CONSTS
     1  FACT       REAL                 CONSTS           34  GS        REAL        ARRAY    IN2
    44  HBW        REAL                 CONSTS           43  HEW       REAL                 CONSTS
     7  IAMAT      INTEGER     ARRAY    CONSTS           57  ID        INTEGER     ARRAY    CONSTS
   250  IGOB       INTEGER     ARRAY    IN2              55  IPHASE    INTEGER              CONSTS
     1  ITMAX      INTEGER              IN7              54  ITMAXI    INTEGER              CONSTS
    53  ITMINI     INTEGER              CONSTS            3  ITP       INTEGER              IN7
    56  JOG        INTEGER              CONSTS           51  KB        INTEGER              CONSTS
   105  KD0        INTEGER              CONSTS          106  KD1       INTEGER              CONSTS
   107  KD2        INTEGER              CONSTS          110  KD3       INTEGER              CONSTS
   111  KD4        INTEGER              CONSTS          112  KD5       INTEGER              CONSTS
   113  KD6        INTEGER              CONSTS          101  KE        INTEGER              CONSTS
   100  KI         INTEGER              CONSTS           77  LQR       INTEGER              CONSTS
   120  NBLKFM     INTEGER              CONSTS          116  NBLKX     INTEGER              CONSTS
   117  NBLKY      INTEGER              CONSTS           50  NBT       INTEGER              CONSTS
        SUBROUTINE INFCOF 74/855 OPT=2 FTN 4.8+587  86/02/05. 13.56.15  PAGE 3
VARIABLES        TYPE          RELOCATION
    47  NBXE       INTEGER              CONSTS          102  NCUT      INTEGER              CONSTS
   114  NELMX      INTEGER              CONSTS          115  NELMY     INTEGER              CONSTS
     4  NGRID      INTEGER              IN7              27  NMATS     INTEGER              IN1
   104  NONE       INTEGER              CONSTS          133  NP        INTEGER
```

```
  134  NPM       INTEGER                              2  NRUN      INTEGER             IN7
   26  NSEAM     INTEGER             IN1             52  NTR       INTEGER             CONSTS
    0  ORF       REAL                IN7            124  PHI       REAL      ARRAY     IN2
  214  PHI1      REAL      ARRAY     IN2              0  PI        REAL                CONSTS
   46  RLIM      REAL                CONSTS         130  SS1       REAL
  131  SS2       REAL                               132  SS3       REAL
    0  S13U      REAL      ARRAY     BK4           6344  S23U      REAL      ARRAY     BK4
23254  S23V      REAL      ARRAY     BK4          14710  S33U      REAL      ARRAY     BK4
31620  S33V      REAL      ARRAY     BK4          40164  S33W      REAL      ARRAY     BK4
   14  THIKNS    REAL      ARRAY     IN5              0  TITLE     REAL      ARRAY     IN1
   24  V         REAL                IN1              2  V1        REAL                CONSTS
    3  V2        REAL                CONSTS           0  XO        REAL      ARRAY     IN5
    4  YO        REAL      ARRAY     IN5             10  ZO        REAL      ARRAY     IN5
FILE NAMES        MODE
       TAPE16    UNFMT               TAPE20    UNFMT
EXTERNALS           TYPE   ARGS
       COEF1               0                           COEF2               0
STATEMENT LABELS
    0  140                                0  150                                 50  200
   54  999
LOOPS  LABEL    INDEX     FROM-TO    LENGTH    PROPERTIES
   21  140      NP         53 57        3B         OPT
   31  150      NP         61 66       12B               NOT INNER
   33  150      NPM        62 66        7B         OPT
COMMON BLOCKS   LENGTH
       BK4       20124
       BK5       26400
       CONSTS       81
       IN1          24
       IN2         196
       IN5          16
       IN7           5
STATISTICS
  PROGRAM LENGTH                  135B        93
  CM LABELED COMMON LENGTH     133376B     46846
         60000B CM USED
```

```
      C
            SUBROUTINE INFILE
      C
      C..THIS SUBROUTINE WILL EITHER INITIALIZE OR LOAD THE MASS
      C  STORAGE FILE CONTAINING RIDE AND CLOSURE VALUES
      C
      C  THE FIRST 4 RECORDS CONSIST OF AVERAGE BLOCK VALUES.
      C     1.  RIDES AND CLOSURES FOR THE X DIRECTION
      C     2.  RIDES AND CLOSURES FOR THE Y DIRECTION
      C     3.  RIDES AND CLOSURES FOR THE Z DIRECTION
      C     4.  MINING CODES FOR ALL THE BLOCKS IN THE GRID
      C
      C  THE REST OF THE FILE CONSISTS OF ONE RECORD FOR THE ELEMENTAL
      C  RIDES AND CLOSURES OF EACH BLOCK IN THE GRID.
      C
      C
            COMMON/BK1/    DUB(3200), DVB(3200), DWB(3200),DWBI(3200),
           #               DE(2100),  INDEX(805),  KODE(25), INDX13(805)
           #               ,NGOB(3200)
            COMMON/CONSTS/ PI, FACT, V1, V2, CONS, COND, CON, IAMAT(28),
           #               HEW, HBW, EW, RLIM, NBXE, NBT, KB, NTR, ITMINI,
           #               ITMAXI, IPHASE, JOG, ID(4,4), LQR, KI, KE, NCUT,
           #               ACC, NONE, KD0, KD1, KD2, KD3, KD4, KD5, KD6
           #               , NELMX, NELMY, NBLKX, NBLKY, NBLKFM
            COMMON/IN7/    ORF, ITMAX, NRUN, ITP, NGRID
            COMMON/IN9/    MKOD(2403), MCKOD(3200), MCEKOD(3200)
      C
            CALL OPENMS(10,INDEX,KI,0)
      C
            IF (ITP .EQ. 0) GO TO 50
              CALL READMS(10,DUB,NBT,1)
              CALL READMS(10,DVB,NBT,2)
              CALL READMS(10,DWB,NBT,3)
              CALL READMS(10,MKOD,KB,4)
            GO TO 999
```

```
           50   DO 100 NR = 1,3
          100     CALL WRITMS(10,DUB,NBT,NR,0)
                CALL WRITMS(10,MKOD,KB,4,0)
                DO 200 NR=5,NTR
          200      CALL WRITMS(10,DE,75,NR,0)
        C
          999 RETURN
              END

    SYMBOLIC REFERENCE MAP (R=1)
ENTRY POINTS
    1  INFILE
VARIABLES        TYPE         RELOCATION
  103  ACC       REAL              CONSTS         6  CON       REAL               CONSTS
    5  COND      REAL              CONSTS         4  CONS      REAL               CONSTS
31000  DE        REAL       ARRAY  BK1            0  DUB       REAL       ARRAY   BK1

    SUBROUTINE INFILE 74/855 OPT=2 FTN 4.8+587  86/02/05. 13.56.15  PAGE 2
VARIABLES        TYPE         RELOCATION
 6200  DVB       REAL       ARRAY  BK1        14400  DWB       REAL       ARRAY   BK1
22600  DWBI      REAL       ARRAY  BK1           45  EW        REAL               CONSTS
    1  FACT      REAL              CONSTS        44  HBW       REAL               CONSTS
   43  HEW       REAL              CONSTS         7  IAMAT     INTEGER    ARRAY   CONSTS
   57  ID        INTEGER    ARRAY  CONSTS     35064  INDEX     INTEGER    ARRAY   BK1
36562  INDX13    INTEGER    ARRAY  BK1           55  IPHASE    INTEGER            CONSTS
    1  ITMAX     INTEGER           IN7           54  ITMAXI    INTEGER            CONSTS
   53  ITMINI    INTEGER           CONSTS         3  ITP       INTEGER            IN7
   56  JOG       INTEGER           CONSTS        51  KB        INTEGER            CONSTS
  105  KD0       INTEGER           CONSTS       106  KD1       INTEGER            CONSTS
  107  KD2       INTEGER           CONSTS       110  KD3       INTEGER            CONSTS
  111  KD4       INTEGER           CONSTS       112  KD5       INTEGER            CONSTS
  113  KD6       INTEGER           CONSTS       101  KE        INTEGER            CONSTS
  100  KI        INTEGER           CONSTS     36531  KODE      INTEGER    ARRAY   BK1
   77  LQR       INTEGER           CONSTS     12743  MCEKOD    INTEGER    ARRAY   IN9
 4543  MCKOD     INTEGER    ARRAY  IN9            0  MKOD      INTEGER    ARRAY   IN9
  120  NBLKFM    INTEGER           CONSTS       116  NBLKX     INTEGER            CONSTS
```

```
  117  NBLKY     INTEGER               CONSTS          50  NBT      INTEGER            CONSTS
   47  NBXE      INTEGER               CONSTS         102  NCUT     INTEGER            CONSTS
  114  NELMX     INTEGER               CONSTS         115  NELMY    INTEGER            CONSTS
40227  NGOB      INTEGER    ARRAY      BK1              4  NGRID    INTEGER            IN7
  104  NONE      INTEGER               CONSTS         116  NR       INTEGER
    2  NRUN      INTEGER               IN7             52  NTR      INTEGER            CONSTS
    0  ORF       REAL                  IN7              0  PI       REAL               CONSTS
   46  RLIM      REAL                  CONSTS           2  V1       REAL               CONSTS
    3  V2        REAL                  CONSTS
EXTERNALS          TYPE    ARGS
       OPENMS                4                             READMS               4
       WRITMS                5
STATEMENT LABELS
   16  50                                 0  100                                   0  200
   33  999
LOOPS  LABEL     INDEX     FROM-TO     LENGTH     PROPERTIES
   17  100       NR          36 37         5B               EXT REFS
   26  200       NR          39 40         5B               EXT REFS
COMMON BLOCKS    LENGTH
       BK1        19735
       CONSTS        81
       IN7            5
       IN9         8803
STATISTICS
  PROGRAM LENGTH                117B       79
  CM LABELED COMMON LENGTH    67720B    28624
       60000B CM USED

       SUBROUTINE INIGOB 74/875 OPT=2 FTN 4.8+670  87/09/01. 13.42.10  PAGE 1
                             SUBROUTINE INIGOB
                      C
                      C..  THIS SUBROUTINE WILL INITIALIZE THE CLOSURE
                      C    VALUES FOR ALL GOB ELEMENTS
                      C
                            COMMON/BK1/    DUB(3200), DVB(3200), DWB(3200),DWBI(3200),
                           #               DE(2100),  INDEX(805),  KODE(25), INDX13(805)
```

```
     #               ,NGOB(3200)
      COMMON/BK2/    PO1(4), PO2(4), PO3(4),
     #               P1XI,   P2XI,   P3XI,
     #               P1ET,   P2ET,   P3ET
      COMMON/BK3/    EN(3,3), IMAX, JMAX
      COMMON/IN1/    TITLE(20), V, E, NSEAM, NMATS
      COMMON/IN2/    ES(28), GS(28), COHES(28),
     #               PHI(28), COHES1(28), PHI1(28), IGOB(28)
      COMMON/IN4/    BW, NBXI, NBET, NSYM, NOSP, IFXS, IFXE, IFYS, IFYE
      COMMON/IN4A/   IEX, EXR(10)
      COMMON/IN5/    XO(4), YO(4), ZO(4), THIKNS(4)
      COMMON/IN7/    ORF, ITMAX, NRUN, ITP, NGRID
      COMMON/IN9/    MKOD(2403), MCKOD(3200), MCEKOD(3200)
      COMMON/CONSTS/ PI, FACT, V1, V2, CONS, COND, CON, IAMAT(28),
     #               HEW, HBW, EW, RLIM, NBXE, NBT, KB, NTR, ITMINI,
     #               ITMAXI, IPHASE, JOG, ID(4,4), LQR, KI, KE, NCUT,
     #               ACC, NONE, KD0, KD1, KD2, KD3, KD4, KD5, KD6
     #               , NELMX, NELMY, NBLKX, NBLKY, NBLKFM
C
      DIMENSION MAT(25), CLOS(75)
C
      REWIND 12
      IREC = 4
C
C..SEAM LOOP
C
      DO 1000 NP = 1,NSEAM
        THICK = THIKNS(NP)
        LPBO = (NP - 1) * NBXE
        IKP = (NP - 1) * NBLKX * NBLKY
C..BLOCK LOOP
C
        DO 900  JP = 1,NBXI
          DO 800 IP = 1,NBET
            LPB = LPBO + (JP - 1) * NBET + IP
            I = 0
```

```
          IGOBFL = 0
C
C..CHECK FOR FINE MESH BLOCK
C
          IF(IP .GE. IFYS .AND. IP .LE. (IFYS+NBLKY-1) .AND.
    #        JP .GE. IFXS .AND. JP .LE. (IFXS+NBLKX-1) GO TO 400
C
          FACTR = 1
          DWBI(LPB) = 0.0
C
C.. IF THE BLOCK IS AN OPENING(27), OR RIGID(28)
C   THEN THE INITIAL CLOSURE REMAINS ZERO.
C
          IF( MCKOD(LPB) .EQ. 27 .OR.
```

SUBROUTINE INIGOB 74/875 OPT=2 FTN 4.8+670 87/09/01. 13.42.10 PAGE 2

```
    #         MCKOD(LPB) .EQ. 28 ) GO TO 750
C
C.. IF THE BLOCK IS A GOB BLOCK THEN PROCEED TO CALCULATION
C
          IF( IGOB( MCKOD(LPB) ) .EQ. 1 ) IGOBFL = 25
          IF( IGOB( MCKOD(LPB) ) .NE. 0 ) GO TO 300
C
C..CHECK TO SEE IF SEAM BLOCK HAS ZERO EXTRACTION
C  IF SO THEN INITIAL CLOSURE REMAINS ZERO
C
         IF( MCEKOD(LPB) .EQ. 10)    GO TO 750
C
          FACTR = EXR( MCEKOD(LPB) ) / ( 1. - EXR( MCEKOD(LPB) ) )
C
C
  300     JLP = (JP - 1) * 5 + 2
          IC = (IP - 1) * 5 + 2
          P3 = PO3(NP) + JLP * P3XI + IC * P3ET
          DWBI(LPB) = (( P3 * THICK) / ES( MCKOD(LPB) ) ) * FACTR
```

```
            GO TO 750
  400       IC = (JP-IFXS) * NBLKY + (IP -IFYS +1) + IKP
            CALL READMS(13,MAT,25,IC)
            IREC = IREC + 1
            DO 500 IC = 1,75
              CLOS(IC) = 0.0
  500       CONTINUE
C
C.. ELEMENT LOOP
C
            DO 700 LP = 1,5
              DO 600 KP = 1,5

                I = I + 1
                INDX = MAT(I)
                IF (IGOB(INDX) .EQ. 0 ) GO TO 600
                  IGOBFL = IGOBFL + 1
                  JLP = (JP-1) * 5 + LP - 1
                  IC = (IP-1) * 5 + KP - 1
                  P3  = PO3(NP) + JLP * P3XI + IC * P3ET
                  PC  = P3 / ES(INDX)
C
C..ERROR EXIT IF PERCENT CLOSURE WAS EQUAL TO SEAM THICKNESS
C
                  IF(PC .LT. 1) GO TO 550
                    PRINT 10000,NP,JP,IP,LP,KP,INDX,ES(INDX)
                    STOP
  550             CONTINUE
                  CLOS(I+50) = PC * THICK
                  DWBI(LPB) = DWBI(LPB) + (.04 * CLOS(I+50))
  600         CONTINUE
  700       CONTINUE
C
C..REWRITE DIRECT ACCESS CLOSURE RECORD
C  IF GOB ELEMENT FOUND IN THE LAST BLOCK
C
```

```
                    IF(IGOBFL .EQ. 0 .OR. ITP .GT. 0) GO TO 750
                      CALL WRITMS(10,CLOS(1),75,IREC,1)

   SUBROUTINE INIGOB 74/875 OPT=2 FTN 4.8+670  87/09/01. 13.42.10  PAGE 3

          C
          C
          C.. IF MAIDEN RUN THEN SET BLOCK CLOSURE TO INITIAL BLOCK CLOSURE
          C
            750       IF( ITP .EQ. 0 ) DWB(LPB) = DWBI(LPB)
          C
                      NGOB(LPB) = IGOBFL
            800     CONTINUE
            900   CONTINUE
           1000 CONTINUE
          10000 FORMAT(//,*  EQUILIBRIUM GOB CLOSURE GREATER THAN*  ,//,
               # *  OR EQUAL TO SEAM THICKNESS.  INCREASE THE VALUE*  ,//,
               # *    OF YOUNGS MODULUS FOR THE GOB MATERIAL* ,//,
               # *  SEAM = *,I10,//,
               # *  BLOCK COLUMN = *,I10,//,
               # *  BLOCK ROW    = *,I10,//,
               # *  ELEMENT COL  = *,I10,//,
               # *  ELEMENT ROW  = *,I10,//,
               # *  MATERIAL TYPE= *,I10,//,
               # *  YOUNGS MODULUS= *,E15.5)
                 RETURN
                 END
   SYMBOLIC REFERENCE MAP (R=1)
ENTRY POINTS
   1  INIGOB
VARIABLES        TYPE          RELOCATION
  103  ACC       REAL                CONSTS        0  BW       REAL                 IN4
  357  CLOS      REAL       ARRAY                 70  COHES    REAL       ARRAY     IN2
  160  COHES1    REAL       ARRAY    IN2           6  CON      REAL                 CONSTS
    5  COND      REAL                CONSTS        4  CONS     REAL                 CONSTS
31000  DE        REAL       ARRAY    BK1           0  DUB      REAL       ARRAY     BK1
```

```
 6200  DVB     REAL     ARRAY  BK1        14400  DWB     REAL     ARRAY  BK1
22600  DWBI    REAL     ARRAY  BK1           25  E       REAL            IN1
    0  EN      REAL     ARRAY  BK3            0  ES      REAL     ARRAY  IN2
   45  EW      REAL            CONSTS         1  EXR     REAL     ARRAY  IN4A
    1  FACT    REAL            CONSTS       316  FACTR   REAL
   34  GS      REAL     ARRAY  IN2           44  HBW     REAL            CONSTS
   43  HEW     REAL            CONSTS       314  I       INTEGER
    7  IAMAT   INTEGER  ARRAY  CONSTS       320  IC      INTEGER
   57  ID      INTEGER  ARRAY  CONSTS         0  IEX     INTEGER         IN4A
    6  IFXE    INTEGER         IN4            5  IFXS    INTEGER         IN4
   10  IFYE    INTEGER         IN4            7  IFYS    INTEGER         IN4
  250  IGOB    INTEGER  ARRAY  IN2          315  IGOBFL  INTEGER
  310  IKP     INTEGER                       11  IMAX    INTEGER         BK3
35064  INDEX   INTEGER  ARRAY  BK1          324  INDX    INTEGER
36562  INDX13  INTEGER  ARRAY  BK1          312  IP      INTEGER
   55  IPHASE  INTEGER         CONSTS       304  IREC    INTEGER
    1  ITMAX   INTEGER         IN7           54  ITMAXI  INTEGER         CONSTS
   53  ITMINI  INTEGER         CONSTS         3  ITP     INTEGER         IN7
  317  JLP     INTEGER                       12  JMAX    INTEGER         BK3

       SUBROUTINE INIGOB 74/875 OPT=2 FTN 4.8+670  87/09/01. 13.42.10  PAGE 4
VARIABLES      TYPE          RELOCATION
   56  JOG     INTEGER         CONSTS       311  JP      INTEGER
   51  KB      INTEGER         CONSTS       105  KD0     INTEGER         CONSTS
  106  KD1     INTEGER         CONSTS       107  KD2     INTEGER         CONSTS
  110  KD3     INTEGER         CONSTS       111  KD4     INTEGER         CONSTS
  112  KD5     INTEGER         CONSTS       113  KD6     INTEGER         CONSTS
  101  KE      INTEGER         CONSTS       100  KI      INTEGER         CONSTS
36531  KODE    INTEGER  ARRAY  BK1          323  KP      INTEGER
  322  LP      INTEGER                      313  LPB     INTEGER
  307  LPBO    INTEGER                       77  LQR     INTEGER         CONSTS
  326  MAT     INTEGER  ARRAY             12743  MCEKOD  INTEGER  ARRAY  IN9
 4543  MCKOD   INTEGER  ARRAY  IN9            0  MKOD    INTEGER  ARRAY  IN9
    2  NBET    INTEGER         IN4          120  NBLKFM  INTEGER         CONSTS
  116  NBLKX   INTEGER         CONSTS       117  NBLKY   INTEGER         CONSTS
   50  NBT     INTEGER         CONSTS        47  NBXE    INTEGER         CONSTS
```

```
    1  NBXI      INTEGER            IN4          102  NCUT      INTEGER              CONSTS
  114  NELMX     INTEGER            CONSTS       115  NELMY     INTEGER              CONSTS
40227  NGOB      INTEGER   ARRAY    BK1            4  NGRID     INTEGER              IN7
   27  NMATS     INTEGER            IN1          104  NONE      INTEGER              CONSTS
    4  NOSP      INTEGER            IN4          305  NP        INTEGER
    2  NRUN      INTEGER            IN7           26  NSEAM     INTEGER              IN1
    3  NSYM      INTEGER            IN4           52  NTR       INTEGER              CONSTS
    0  ORF       REAL               IN7          325  PC        REAL
  124  PHI       REAL      ARRAY    IN2          214  PHI1      REAL      ARRAY      IN2
    0  PI        REAL               CONSTS         0  PO1       REAL      ARRAY      BK2
    4  PO2       REAL      ARRAY    BK2           10  PO3       REAL      ARRAY      BK2
   17  P1ET      REAL               BK2           14  P1XI      REAL                 BK2
   20  P2ET      REAL               BK2           15  P2XI      REAL                 BK2
  321  P3        REAL                             21  P3ET      REAL                 BK2
   16  P3XI      REAL               BK2           46  RLIM      REAL                 CONSTS
  306  THICK     REAL                             14  THIKNS    REAL      ARRAY      IN5
    0  TITLE     REAL      ARRAY    IN1           24  V         REAL                 IN1
    2  V1        REAL               CONSTS         3  V2        REAL                 CONSTS
    0  XO        REAL      ARRAY    IN5            4  YO        REAL      ARRAY      IN5
   10  ZO        REAL      ARRAY    IN5

FILE NAMES      MODE
       OUTPUT   FMT                 TAPE12

EXTERNALS        TYPE    ARGS
       READMS             4                         WRITMS                5

STATEMENT LABELS
   60   300                            75  400                              0  500
  145   550                           151  600                              0  700
  161   750                             0  800                              0  900
    0   1000                          231  10000   FMT

LOOPS  LABEL   INDEX    FROM-TO   LENGTH   PROPERTIES
    7  1000    NP        34 124    167B             EXT REFS  NOT INNER
   16  900     JP        40 123    155B             EXT REFS  NOT INNER
```

```
  24  800      IP          41 122      145B                EXT REFS  NOT INNER
 107  500      IC          81 83         2B    INSTACK
 121  700      LP          87 108       33B                EXT REFS  NOT INNER
 124  600      KP          88 107       26B                EXT REFS

      SUBROUTINE INIGOB 74/875 OPT=2 FTN 4.8+670  87/09/01. 13.42.10  PAGE 5
COMMON BLOCKS   LENGTH
      BK1        19735
      BK2           18
      BK3           11
      IN1           24
      IN2          196
      IN4            9
      IN4A          11
      IN5           16
      IN7            5
      IN9         8803
      CONSTS        81

STATISTICS
  PROGRAM LENGTH                  472B     314
  SCM LABELED COMMON LENGTH    70355B    28909
        60000B SCM USED

      SUBROUTINE OFSIM 74/875 OPT=2 FTN 4.8+670  87/09/01. 13.42.10  PAGE 1

                  SUBROUTINE OFSIM(NOP)
            C
            C  THIS ROUTINE COMPUTES THE OFF-SEAM DISPLACEMENTS AND STRESSES.
            C
                  COMMON/BK1/    DUB(3200), DVB(3200), DWB(3200),DWBI(3200),
                 #               DE(2100),  INDEX(805),  KODE(25), INDX13(805)
                 #               ,NGOB(3200)
                  COMMON/BK2/    S11PE,S12PE,S13PE,S22PE,S23PE,S33PE,UPE,VPE,WPE,
                 #               UX,VY,WZ,SXX,SXY,SXZ,SYY,SYZ,SZZ
                  COMMON/BK3/    EN(3,3), IMAX, JMAX
```

```
      COMMON/BK4/    OS11U(25),OS12U(25),OS13U(25),
     #               OS22U(25),OS23U(25),OS33U(25),OS11V(25),
     #               OS12V(25),OS22V(25),OS23V(25),OS33V(25),
     #               OS11W(25),OS12W(25),OS22W(25),OS33W(25),
     #               OD1U(25),OD2U(25),OD3U(25),OD2V(25),
     #               OD3V(25),OD1W(25),OD2W(25),OD3W(25)
      COMMON/BK7/    XI, ET, ZT, SEG, DERI01, DERI02, DERI03, DERI04,
     #               DERI05, DERI06, DERI07, DERI08, DERI09, DERI10,
     #               DERI11, DERI12, DERI13, DERI14, DERI15, DERI16,
     #               DERI17, DERI18, DERI19
      COMMON/CONSTS/ PI, FACT, V1, V2, CONS, COND, CON, IAMAT(28),
     #               HEW, HBW, EW, RLIM, NBXE, NBT, KB, NTR, ITMINI,
     #               ITMAXI, IPHASE, JOG, ID(4,4), LQR, KI, KE, NCUT,
     #               ACC, NONE, KD0, KD1, KD2, KD3, KD4, KD5, KD6
     #               , NELMX, NELMY, NBLKX, NBLKY, NBLKFM
      COMMON/IN1/    TITLE(20), V, E, NSEAM, NMATS
      COMMON/IN2/    ES(28), GS(28), COHES(28),
     #               PHI(28), COHES1(28), PHI1(28), IGOB(28)
      COMMON/IN3/    AXX, AXY, AXZ, AYY, AYZ, AZZ,
     #               BXX, BXY, BXZ, BYY, BYZ, BZZ
      COMMON/IN4/    BW, NBXI, NBET, NSYM, NOSP, IFXS, IFXE, IFYS, IFYE
      COMMON/IN4A/   IEX, EXR(10)
      COMMON/IN5/    XO(4), YO(4), ZO(4), THIKNS(4)
      DIMENSION DISTRES(9),USBE(9), MAT4(25), MAT5(25)
      EQUIVALENCE (DISTRES,UX),(S11PE,USBE(1))
C
C  LOOP FOR EACH OFF SEAM PLANE
C
      DO 350 NP = 1, NOP
C
C  GRID DETAILS FOR THIS OFF-SEAM PLANE
C
        READ 10000, XOP, YOP, ZOP, B1, B2, N1, N2
        READ 10100, EN1X, EN2X, EN3X, EN1Y, EN2Y, EN3Y, EN1Z, EN2Z, EN3Z
        PRINT 10200, NP
        PRINT 10300, XOP, YOP, ZOP, B1, B2, N1, N2
```

```
      PRINT 10400, EN1X, EN2X, EN3X, EN1Y, EN2Y, EN3Y, EN1Z, EN2Z,EN3Z
      PRINT 10500
C
C..CHECK TO SEE IF THE OFFSEAM PLANE IS WITHIN A SEAM
C
      DO 100 NS = 1,NSEAM
        IF(ZOP .GT. ZO(NS) .OR. ZOP .LT. (ZO(NS)-THIKNS(NS)))
     #    GO TO 100
        PRINT 10700,NP,NS
        GO TO 350
  100 CONTINUE
```

SUBROUTINE OFSIM 74/875 OPT=2 FTN 4.8+670 87/09/01. 13.42.10 PAGE 2

```
C
      H1 = B1 / 2.
      H2 = B2 / 2.
      NE = 0
      N12 = N1 * N2
C
C  RATE OF CHANGE OF COORDINATES-- XI,ET,ZT (LOCAL SYSTEM AT ANY
C  SEAM PLANE) AT  THIS OFF-SEAM PLANE
C
      HXI1 = H1 * (EN1X * EN(1,1) + EN1Y * EN(1,2) + EN1Z * EN(1,3))
      HET1 = H1 * (EN1X * EN(2,1) + EN1Y * EN(2,2) + EN1Z * EN(2,3))
      HZT1 = H1 * (EN1X * EN(3,1) + EN1Y * EN(3,2) + EN1Z * EN(3,3))
      HXI2 = H2 * (EN2X * EN(1,1) + EN2Y * EN(1,2) + EN2Z * EN(1,3))
      HET2 = H2 * (EN2X * EN(2,1) + EN2Y * EN(2,2) + EN2Z * EN(2,3))
      HZT2 = H2 * (EN2X * EN(3,1) + EN2Y * EN(3,2) + EN2Z * EN(3,3))
C
C..LOOP FOR ALL POINTS WITHIN EACH OFSEAM PLANE
C
      DO 335 JP = 1, N1
        EJ = FLOAT(2 * JP - 1)
        DO 335 IP = 1, N2
          EI = FLOAT(2 * IP - 1)
```

```
            NE = NE + 1
C
C  INITIALIZE THE DISPLACEMENT AND STRESS SUMPS AND THE NEIGHBORHOOD
C  DIRECTORY
C
            DO 120 M = 1, 9
              USBE(M) = 0.0
  120       CONTINUE
C
C..LOOP FOR ALL SEAMS
C
            DO 330 NS = 1, NSEAM
              ILOWER = 0
              IF(ZOP .GT. ZO(NS)) ILOWER = 1
              NR = (NS-1) * NBXE
              NRFM = (NS-1) * NBLKFM
C
C  COORDINATE TRANSFORMATIONS- GLOBAL TO LOCAL (AT THE SEAM NS) SYSTEM
C
              X = XOP - XO(NS)
              Y = YOP - YO(NS)
              Z = ZOP - ZO(NS)
              XIP = X *EN(1,1)+Y *EN(1,2)+Z *EN(1,3)+EJ *HXI1+EI *HXI2
              ETP=X*EN(2,1)+Y*EN(2,2)+Z*EN(2,3)+EJ*HET1+EI*HET2
              ZTP=X*EN(3,1)+Y*EN(3,2)+Z*EN(3,3)+EJ*HZT1+EI*HZT2
              ZT=ZTP
C
C..CHANGES MADE TO DETERMINE THE INITIAL CLOSURE
C  OF THE CLOSEST BLOCK TO THE OFF-SEAM BLOCK
C
              CCB = 0
              GC   = 0
C
C..CHECK TO SEE IF POINT IS WITHIN THE GRID MODEL
C
```

```
                  IF (XIP .LT. XO(NS)) GO TO 130
                  IF (XIP .GT. XO(NS) + BW * NBXI) GO TO 130
                  IF (ETP .LT. YO(NS)) GO TO 130
                  IF (ETP .GT. YO(NS) + BW * NBET) GO TO 130
                    IXBLK = (XIP - XO(NS)) / BW + 1
                    IYBLK = (ETP - YO(NS)) / BW + 1
                    LXB   = NR + (IXBLK - 1) * NBET + IYBLK
                    CCB  = DWBI(LXB)
                    IF (NGOB(LXB) .GT. 0)
     #                GC   = 25 * DWBI(LXB) / NGOB(LXB)
                    IF(IXBLK .LT. IFXS .OR.
     #                 IXBLK .GT. (IFXS+NBLKX-1) .OR.
     #                 IYBLK .LT. IFYS .OR.
     #                 IYBLK .GT. (IFYS+NBLKY-1)) GO TO 130
                      XOB    = XO(NS) + (IXBLK - 1) * BW
                      YOB    = YO(NS) + (IYBLK - 1) * BW
                      IXEL   = (XIP - XOB) / EW + 1
                      IYEL   = (ETP - YOB) / EW + 1
                      LXBREC = NRFM + (IXBLK-IFXS) * NBLKY+(IYBLK-IFYS)+1
                      CALL READMS(13,MAT4,25,LXBREC)
                      IMAT4 = MAT4(5 * IXEL + IYEL -5)
                      EC     = GC * IGOB(IMAT4)
  130             CONTINUE
C
C..LOOP FOR ALL BLOCKS WITHIN EACH SEAM
C
                  DO 325 JS=1,NBXI
                    XIB=XIP-2*(JS-1)*HBW
                    DO 325 IS=1,NBET
                      ETB=ETP-2*(IS-1)*HBW
                      LSB = NR + (JS-1) * NBET + IS
                      GC3 = 0
                      IF(NGOB(LSB) .GT. 0)
     #                  GC3 = 25. * DWBI(LSB) / NGOB(LSB)
```

```
                 SEG=HBW
                 NL=1
                 NK=NL
C
C..CHECK FOR A COARSE MESH BLOCK
C
                 IF(JS .LT. IFXS .OR.
    #               JS .GT. (IFXS+NBLKX-1) .OR.
    #               IS .LT. IFYS .OR.
    #               IS .GT. (IFYS+NBLKY-1)) GO TO 140
C
                 R=SQRT((XIB-HBW)**2+(ETB-HBW)**2+ZT**2)
C
C  CHECK IF THE INFLUENCE OF EACH ELEMENT OF THIS BLOCK NEED BE
C  CONSIDERED INDIVIDUALLY, IF YES- THEN SET SEG TO
C  HALF THE ELEMENT WIDTH AND SET INDICIES TO CALCULATE COEFFICIENTS
C  FOR ALL ELEMENTS IN THE BLOCK
C
                 IF(R.GT.RLIM) GO TO 140
                 SEG=HEW
                 NL=5
                 NK=NL
  140            CONTINUE
```

SUBROUTINE OFSIM 74/875 OPT=2 FTN 4.8+670 87/09/01. 13.42.10 PAGE 4

```
                 NUM = 0
                 DO 145 LS=1,NL
                   XI=XIB-(2*LS-1)*SEG
                   DO 145 KS=1,NK
                     ET=ETB-(2*KS-1)*SEG
                     NUM=NUM+1
                     CALL DERIVE
C
C  STRESS COEFFICIENTS--
C
```

```
                    OS11U(NUM)=ZT*DERI10-2.*DERI06
                    OS12U(NUM)=ZT*DERI11-V1*DERI08
                    OS13U(NUM)=V*DERI04-V1*DERI09+ZT*DERI12
                    OS22U(NUM)=ZT*DERI13-2.*V*DERI06
                    OS23U(NUM)=V*DERI05+ZT*DERI14
                    OS33U(NUM)=ZT*DERI15
                    OS11V(NUM)=ZT*DERI11-2.*V*DERI08
                    OS12V(NUM)=ZT*DERI13-V1*DERI06
                    OS22V(NUM)=ZT*DERI16-2.*DERI08
                    OS23V(NUM)=V*DERI07-V1*DERI09+ZT*DERI17
                    OS33V(NUM)=ZT*DERI18
                    OS11W(NUM)=V2*DERI04-2.*V*DERI09+ZT*DERI12
                    OS12W(NUM)=V2*DERI05+ZT*DERI14
                    OS22W(NUM)=V2*DERI07-2.*V*DERI09+ZT*DERI17
                    OS33W(NUM)=ZT*DERI19-DERI09
C
C  DISPLACEMENT COEFFICIENTS--
C
                    OD1U(NUM)=2.*V1*DERI03-ZT*DERI04
                    OD2U(NUM)=-ZT*DERI05
                    OD3U(NUM)=V2*DERI01-ZT*DERI06
                    OD2V(NUM)=2.*V1*DERI03-ZT*DERI07
                    OD3V(NUM)=V2*DERI02-ZT*DERI08
                    OD1W(NUM)=-V2*DERI01-ZT*DERI06
                    OD2W(NUM)=-V2*DERI02-ZT*DERI08
                    OD3W(NUM)=2.*V1*DERI03-ZT*DERI09
  145             CONTINUE
C
C..IF ONLY BLOCK TO BLOCK COEFFICIENTS WERE CALCULATED
C  SKIP TO AVERAGE BLOCK INFLUENCE SECTION
C
                  IF( NL .EQ. 1 ) GO TO 300
                  LSBREC = NRFM + (JS-IFXS) * NBLKY + (IS-IFYS) + 1
                  CALL READMS(13,MAT5,25,LSBREC)
C
C  RETRIEVE THE ELEMENTAL RIDE AND CLOSURE VALUES FOR THIS BLOCK
```

```
C
                NRC = LSBREC + 4
                CALL READMS(10,DE,75,NRC)
                DO 290 NUM=1,25
                  EC3 = GC3 * IGOB(MAT5(NUM))
                  LSEU = NUM
                  LSEV = LSEU + 25
                  LSEW = LSEV + 25
C
C..REVISED ELEMENT CLOSURE
C
```

SUBROUTINE OFSIM 74/875 OPT=2 FTN 4.8+670 87/09/01. 13.42.10 PAGE 5

```
                  RDE = DE(LSEW) - EC3
                  IF(EC3 .EQ. 0) GO TO 250
C
C..INFLUENCE OF GOB ELEMENT
C
                IF(ILOWER .EQ. 1) GO TO 240
C
C..GOB IN A HIGHER SEAM
C
                IF(RDE .LT. 0) RDE = 0
                GO TO 250
C
C..GOB IN A LOWER SEAM
C
  240           IF(DE(LSEW) .LT. EC3) RDE = 2*DE(LSEW)
C
C..IF GOB IS STRESSED TO DE > EC3 THEN
C  RDE IS 2 * EC3 + (DE-EC3)
C  THEREFORE
C

                IF(DE(LSEW) .GE. EC3) RDE = DE(LSEW) + EC3
```

```
  250         CONTINUE
C
C
C  INFLUENCE OF EACH ELEMENT OF THIS BLOCK CONSIDERED INDIVIDUALLY
C
               S11PE = S11PE + OS11U(NUM) * DE(LSEU)
     #                       + OS11V(NUM) * DE(LSEV)
     #                       + OS11W(NUM) * (DE(LSEW)-EC)
               S12PE = S12PE + OS12U(NUM) * DE(LSEU)
     #                       + OS12V(NUM) * DE(LSEV)
     #                       + OS12W(NUM) * (DE(LSEW)-EC)
               S13PE = S13PE + OS13U(NUM) * DE(LSEU)
     #                       + OS23U(NUM) * DE(LSEV)
     #                       + OS33U(NUM) * (DE(LSEW)-EC)
               S22PE = S22PE + OS22U(NUM) * DE(LSEU)
     #                       + OS22V(NUM) * DE(LSEV)
     #                       + OS22W(NUM) * (DE(LSEW)-EC)
               S23PE = S23PE + OS23U(NUM) * DE(LSEU)
     #                       + OS23V(NUM) * DE(LSEV)
     #                       + OS33V(NUM) * (DE(LSEW)-EC)
               S33PE = S33PE + OS33U(NUM) * DE(LSEU)
     #                       + OS33V(NUM) * DE(LSEV)
     #                       + OS33W(NUM) * (DE(LSEW)-EC)
               UPE = UPE + OD1U(NUM) * DE(LSEU)
     #                   + OD2U(NUM) * DE(LSEV)
     #                   + OD1W(NUM) * DE(LSEW)
               VPE = VPE + OD2U(NUM) * DE(LSEU)
     #                   + OD2V(NUM) * DE(LSEV)
     #                   + OD2W(NUM) * DE(LSEW)
               WPE = WPE + OD3U(NUM) * DE(LSEU)
     #                   + OD3V(NUM) * DE(LSEV)
     #                   + OD3W(NUM) * RDE
  290         CONTINUE
              GO TO 325
C
```

```
C  ONLY AVERAGE INFLUENCE OF THIS BLOCK CONSIDERED
C
  300             NUM=1
C
C..REVISED BLOCK CLOSURE
C
                  RDWB = DWB(LSB)
                  IF(NGOB(LSB) .EQ. 0) GO TO 320
                  IF(ILOWER .EQ. 1)   GO TO 310
C
C..GOB IN A HIGHER SEAM
C
                  RDWB = DWB(LSB) - DWBI(LSB)
                  IF(RDWB .LT. 0) RDWB = 0
                  GO TO 320
C
C..GOB IN A LOWER SEAM
C
  310             IF(DWB(LSB).LT.DWBI(LSB)) RDWB = 2* DWB(LSB)
C
C..IF DWB > DWBI THEN RDWB IS 2*DWBI + (DWB-DWBI)
C  THEREFORE
                  IF(DWB(LSB).GE.DWBI(LSB)) RDWB = DWB(LSB) + DWBI(LSB)
  320             CONTINUE
                  S11PE = S11PE + OS11U(NUM) * DUB(LSB)
     #                          + OS11V(NUM) * DVB(LSB)
     #                          + OS11W(NUM) * (DWB(LSB)-CCB)
                  S12PE = S12PE + OS12U(NUM) * DUB(LSB)
     #                          + OS12V(NUM) * DVB(LSB)
     #                          + OS12W(NUM) * (DWB(LSB)-CCB)
                  S13PE = S13PE + OS13U(NUM) * DUB(LSB)
     #                          + OS23U(NUM) * DVB(LSB)
     #                          + OS33U(NUM) * (DWB(LSB)-CCB)
                  S22PE = S22PE + OS22U(NUM) * DUB(LSB)
```

```
     #                        + OS22V(NUM) * DVB(LSB)
     #                        + OS22W(NUM) * (DWB(LSB)-CCB)
                S23PE = S23PE + OS23U(NUM) * DUB(LSB)
     #                        + OS23V(NUM) * DVB(LSB)
     #                        + OS33V(NUM) * (DWB(LSB)-CCB)
                S33PE = S33PE + OS33U(NUM) * DUB(LSB)
     #                        + OS33V(NUM) * DVB(LSB)
     #                        + OS33W(NUM) * (DWB(LSB)-CCB)
                UPE = UPE + OD1U(NUM) * DUB(LSB)
     #                    + OD2U(NUM) * DVB(LSB)
     #                    + OD1W(NUM) * DWB(LSB)
                VPE = VPE + OD2U(NUM) * DUB(LSB)
     #                    + OD2V(NUM) * DVB(LSB)
     #                    + OD2W(NUM) * DWB(LSB)
                WPE = WPE + OD3U(NUM) * DUB(LSB)
     #                    + OD3V(NUM) * DVB(LSB)
     #                    + OD3W(NUM) * RDWB
  325         CONTINUE
  330       CONTINUE
C
C  GLOBAL COORDINATES AND GLOBAL COMPONENTS OF DISPLACEMENTS AND
C  STRESSES AT THIS OFF-SEAM LOCATION
C
```

SUBROUTINE OFSIM 74/875 OPT=2 FTN 4.8+670 87/09/01. 13.42.10 PAGE 7

```
            X = XOP + EJ * EN1X * H1 + EI * EN2X * H2
            Y = YOP + EJ * EN1Y * H1 + EI * EN2Y * H2
            Z = ZOP + EJ * EN1Z * H1 + EI * EN2Z * H2
C
            UX = ( EN(1,1) * UPE + EN(2,1) * VPE + EN(3,1) * WPE) * COND
            VY = ( EN(1,2) * UPE + EN(2,2) * VPE + EN(3,2) * WPE) * COND
            WZ = ( EN(1,3) * UPE + EN(2,3) * VPE + EN(3,3) * WPE) * COND
C
            SXX = ( EN(1,1) ** 2 * S11PE +
     #              EN(2,1) ** 2 * S22PE +
```

```
#               EN(3,1) ** 2 * S33PE +
#               2. * EN(1,1) * EN(2,1) * S12PE +
#               2. * EN(1,1) * EN(3,1) * S13PE +
#               2. * EN(2,1) * EN(3,1) * S23PE
#             ) * CONS + AXX - BXX * Z
        SXY = ( EN(1,1) * EN(1,2) * S11PE +
#               EN(2,1) * EN(2,2) * S22PE +
#               EN(3,1) * EN(3,2) * S33PE +
#               ( EN(1,1) * EN(2,2) + EN(2,1) * EN(1,2) ) * S12PE +
#               ( EN(1,1) * EN(3,2) + EN(3,1) * EN(1,2) ) * S13PE +
#               ( EN(2,1) * EN(3,2) + EN(3,1) * EN(2,2) ) * S23PE
#             ) * CONS + AXY - BXY * Z
        SXZ = ( EN(1,1) * EN(1,3) * S11PE +
#               EN(2,1) * EN(2,3) * S22PE +
#               EN(3,1) * EN(3,3) * S33PE +
#               ( EN(1,1) * EN(2,3) + EN(2,1) * EN(1,3) ) * S12PE +
#               ( EN(1,1) * EN(3,3) + EN(3,1) * EN(1,3) ) * S13PE +
#               ( EN(2,1) * EN(3,3) + EN(3,1) * EN(2,3) ) * S23PE
#             ) * CONS + AXZ - BXZ * Z
        SYY = ( EN(1,2) ** 2 * S11PE +
#               EN(2,2) ** 2 * S22PE +
#               EN(3,2) ** 2 * S33PE +
#               2. * EN(1,2) * EN(2,2) * S22PE +
#               2. * EN(1,1) * EN(3,2) * S13PE +
#               2. * EN(2,2) * EN(3,2) * S23PE
#             ) * CONS + AYY - BYY * Z
        SYZ = ( EN(1,2) * EN(1,3) * S11PE +
#               EN(2,2) * EN(2,3) * S22PE +
#               EN(3,2) * EN(3,3) * S33PE +
#               ( EN(1,2) * EN(2,3) + EN(2,3) * EN(1,3) ) * S12PE +
#               ( EN(1,2) * EN(3,3) + EN(3,2) * EN(1,3) ) * S13PE +
#               ( EN(2,2) * EN(3,3) + EN(3,2) * EN(2,3) ) * S23PE
#             ) * CONS + AYZ - BYZ * Z
        SZZ = ( EN(1,3) ** 2 * S11PE +
#               EN(2,3) ** 2 * S22PE +
#               EN(3,3) ** 2 * S33PE +
```

```
      #              2. * EN(1,3) * EN(2,3) * S12PE +
      #              2. * EN(1,3) * EN(3,3) * S13PE +
      #              2. * EN(2,3) * EN(3,3) * S23PE
      #            ) * CONS + AZZ - BZZ * Z
C
C  OUTPUT THE RESULTS
C
           PRINT 10600,NE,X,Y,Z,(DISTRES(I),I=1,9)
  335   CONTINUE
  350 CONTINUE
C

SUBROUTINE OFSIM 74/875 OPT=2 FTN 4.8+670  87/09/01. 13.42.10  PAGE 8

C  FORMAT STATEMENTS
C
10000 FORMAT(5F12.3,2I8)
10100 FORMAT(9F8.5)
10200 FORMAT(*1*///*  OFF-SEAM PLANE NO.                 --*,I4)
10300 FORMAT(/*  LOCATION OF LOCAL ORIGIN AND GRIDWORK DETAILS --*//
     1 *              GRID ORIGIN AT                    BLOCK WIDTHS
     2  NO. OF BLOCKS*/
     3,*        XOP          YOP          ZOP          B1          B2
     4  N1      N2*,/5F12.3,2I8)
10400 FORMAT(/*  ORIENTATION OF THE OFF-SEAM PLANE --*//*       EN1X
     1     EN2X        EN3X        EN1Y        EN2Y        EN3Y        E
     2N1Z        EN2Z        EN3Z*/,9F12.5)
10500 FORMAT(//*       N       X          Y          Z         UX         UY
     1      UZ       SXX        SXY        SXZ        SYY        SYZ
     2  SZZ */)
10600 FORMAT(I8,3F10.2,3F10.6,6F10.2)
10700 FORMAT(* PLANE *,I3,* IS WITHIN SEAM *,I3)
C
C
      RETURN
```

```
                   END

      SYMBOLIC REFERENCE MAP (R=1)
ENTRY POINTS
    3  OFSIM
VARIABLES          TYPE          RELOCATION
  103  ACC         REAL              CONSTS        0  AXX       REAL                 IN3
    1  AXY         REAL              IN3           2  AXZ       REAL                 IN3
    3  AYY         REAL              IN3           4  AYZ       REAL                 IN3
    5  AZZ         REAL              IN3           0  BW        REAL                 IN4
    6  BXX         REAL              IN3           7  BXY       REAL                 IN3
   10  BXZ         REAL              IN3          11  BYY       REAL                 IN3
   12  BYZ         REAL              IN3          13  BZZ       REAL                 IN3
 1443  B1          REAL                         1444  B2        REAL
 1511  CCB         REAL                           70  COHES     REAL       ARRAY     IN2
  160  COHES1      REAL       ARRAY  IN2           6  CON       REAL                 CONSTS
    5  COND        REAL              CONSTS        4  CONS      REAL                 CONSTS
31000  DE          REAL       ARRAY  BK1           4  DERI01    REAL                 BK7
    5  DERI02      REAL              BK7           6  DERI03    REAL                 BK7
    7  DERI04      REAL              BK7          10  DERI05    REAL                 BK7
   11  DERI06      REAL              BK7          12  DERI07    REAL                 BK7
   13  DERI08      REAL              BK7          14  DERI09    REAL                 BK7
   15  DERI10      REAL              BK7          16  DERI11    REAL                 BK7
   17  DERI12      REAL              BK7          20  DERI13    REAL                 BK7
   21  DERI14      REAL              BK7          22  DERI15    REAL                 BK7
   23  DERI16      REAL              BK7          24  DERI17    REAL                 BK7
   25  DERI18      REAL              BK7          26  DERI19    REAL                 BK7
   11  DISTRES     REAL       ARRAY  BK2           0  DUB       REAL       ARRAY     BK1
 6200  DVB         REAL       ARRAY  BK1       14400  DWB       REAL       ARRAY     BK1
22600  DWBI        REAL       ARRAY  BK1          25  E         REAL                 IN1

      SUBROUTINE OFSIM 74/875 OPT=2 FTN 4.8+670  87/09/01. 13.42.10  PAGE 9
VARIABLES          TYPE          RELOCATION
 1524  EC          REAL                         1543  EC3       REAL
 1476  EI          REAL                         1474  EJ        REAL
    0  EN          REAL       ARRAY  BK3        1447  EN1X      REAL
```

1452	EN1Y	REAL			1455	EN1Z	REAL		
1450	EN2X	REAL			1453	EN2Y	REAL		
1456	EN2Z	REAL			1451	EN3X	REAL		
1454	EN3Y	REAL			1457	EN3Z	REAL		
0	ES	REAL	ARRAY	IN2	1	ET	REAL		BK7
1530	ETB	REAL			1507	ETP	REAL		
45	EW	REAL		CONSTS	1	EXR	REAL	ARRAY	IN4A
1	FACT	REAL		CONSTS	1512	GC	REAL		
1532	GC3	REAL			34	GS	REAL	ARRAY	IN2
44	HBW	REAL		CONSTS	1466	HET1	REAL		
1471	HET2	REAL			43	HEW	REAL		CONSTS
1465	HXI1	REAL			1470	HXI2	REAL		
1467	HZT1	REAL			1472	HZT2	REAL		
1461	H1	REAL			1462	H2	REAL		
1551	I	INTEGER			7	IAMAT	INTEGER	ARRAY	CONSTS
57	ID	INTEGER	ARRAY	CONSTS	0	IEX	INTEGER		IN4A
6	IFXE	INTEGER		IN4	5	IFXS	INTEGER		IN4
10	IFYE	INTEGER		IN4	7	IFYS	INTEGER		IN4
250	IGOB	INTEGER	ARRAY	IN2	1500	ILOWER	INTEGER		
1523	IMAT4	INTEGER			11	IMAX	INTEGER		BK3
35064	INDEX	INTEGER	ARRAY	BK1	36562	INDX13	INTEGER	ARRAY	BK1
1475	IP	INTEGER			55	IPHASE	INTEGER		CONSTS
1527	IS	INTEGER			54	ITMAXI	INTEGER		CONSTS
53	ITMINI	INTEGER		CONSTS	1513	IXBLK	INTEGER		
1520	IXEL	INTEGER			1514	IYBLK	INTEGER		
1521	IYEL	INTEGER			12	JMAX	INTEGER		BK3
56	JOG	INTEGER		CONSTS	1473	JP	INTEGER		
1525	JS	INTEGER			51	KB	INTEGER		CONSTS
105	KD0	INTEGER		CONSTS	106	KD1	INTEGER		CONSTS
107	KD2	INTEGER		CONSTS	110	KD3	INTEGER		CONSTS
111	KD4	INTEGER		CONSTS	112	KD5	INTEGER		CONSTS
113	KD6	INTEGER		CONSTS	101	KE	INTEGER		CONSTS
100	KI	INTEGER		CONSTS	36531	KODE	INTEGER	ARRAY	BK1
1540	KS	INTEGER			77	LQR	INTEGER		CONSTS
1537	LS	INTEGER			1531	LSB	INTEGER		
1541	LSBREC	INTEGER			1544	LSEU	INTEGER		

```
 1545  LSEV      INTEGER                            1546  LSEW      INTEGER
 1515  LXB       INTEGER                            1522  LXBREC    INTEGER
 1477  M         INTEGER                            1552  MAT4      INTEGER   ARRAY
 1603  MAT5      INTEGER   ARRAY                       2  NBET      INTEGER            IN4
  120  NBLKFM    INTEGER             CONSTS          116  NBLKX     INTEGER            CONSTS
  117  NBLKY     INTEGER             CONSTS           50  NBT       INTEGER            CONSTS
   47  NBXE      INTEGER             CONSTS            1  NBXI      INTEGER            IN4
  102  NCUT      INTEGER             CONSTS         1463  NE        INTEGER
  114  NELMX     INTEGER             CONSTS          115  NELMY     INTEGER            CONSTS
40227  NGOB      INTEGER   ARRAY     BK1            1534  NK        INTEGER
 1533  NL        INTEGER                              27  NMATS     INTEGER            IN1
  104  NONE      INTEGER             CONSTS            0  NOP       INTEGER             F.P.
    4  NOSP      INTEGER             IN4            1437  NP        INTEGER
 1501  NR        INTEGER                            1542  NRC       INTEGER
 1502  NRFM      INTEGER                            1460  NS        INTEGER
   26  NSEAM     INTEGER             IN1               3  NSYM      INTEGER            IN4
   52  NTR       INTEGER             CONSTS         1536  NUM       INTEGER
 1445  N1        INTEGER                            1464  N12       INTEGER

       SUBROUTINE OFSIM 74/875 OPT=2 FTN 4.8+670  87/09/01. 13.42.10  PAGE 10
VARIABLES         TYPE          RELOCATION
 1446  N2        INTEGER                             567  OD1U      REAL      ARRAY    BK4
  764  OD1W      REAL      ARRAY     BK4             620  OD2U      REAL      ARRAY    BK4
  702  OD2V      REAL      ARRAY     BK4            1015  OD2W      REAL      ARRAY    BK4
  651  OD3U      REAL      ARRAY     BK4             733  OD3V      REAL      ARRAY    BK4
 1046  OD3W      REAL      ARRAY     BK4               0  OS11U     REAL      ARRAY    BK4
  226  OS11V     REAL      ARRAY     BK4             423  OS11W     REAL      ARRAY    BK4
   31  OS12U     REAL      ARRAY     BK4             257  OS12V     REAL      ARRAY    BK4
  454  OS12W     REAL      ARRAY     BK4              62  OS13U     REAL      ARRAY    BK4
  113  OS22U     REAL      ARRAY     BK4             310  OS22V     REAL      ARRAY    BK4
  505  OS22W     REAL      ARRAY     BK4             144  OS23U     REAL      ARRAY    BK4
  341  OS23V     REAL      ARRAY     BK4             175  OS33U     REAL      ARRAY    BK4
  372  OS33V     REAL      ARRAY     BK4             536  OS33W     REAL      ARRAY    BK4
  124  PHI       REAL      ARRAY     IN2             214  PHI1      REAL      ARRAY    IN2
    0  PI        REAL                CONSTS         1535  R         REAL
 1547  RDE       REAL                               1550  RDWB      REAL
```

```
  46  RLIM      REAL               CONSTS        3  SEG      REAL               BK7
  14  SXX       REAL               BK2          15  SXY      REAL               BK2
  16  SXZ       REAL               BK2          17  SYY      REAL               BK2
  20  SYZ       REAL               BK2          21  SZZ      REAL               BK2
   0  S11PE     REAL               BK2           1  S12PE    REAL               BK2
   2  S13PE     REAL               BK2           3  S22PE    REAL               BK2
   4  S23PE     REAL               BK2           5  S33PE    REAL               BK2
  14  THIKNS    REAL      ARRAY    IN5           0  TITLE    REAL     ARRAY     IN1
   6  UPE       REAL               BK2           0  USBE     REAL     ARRAY     BK2
  11  UX        REAL               BK2          24  V        REAL               IN1
   7  VPE       REAL               BK2          12  VY       REAL               BK2
   2  V1        REAL               CONSTS        3  V2       REAL               CONSTS
  10  WPE       REAL               BK2          13  WZ       REAL               BK2
1503  X         REAL                             0  XI       REAL               BK7
1526  XIB       REAL                          1506  XIP      REAL
   0  XO        REAL      ARRAY    IN5        1516  XOB      REAL
1440  XOP       REAL                          1504  Y        REAL
   4  YO        REAL      ARRAY    IN5        1517  YOB      REAL
1441  YOP       REAL                          1505  Z        REAL
  10  ZO        REAL      ARRAY    IN5        1442  ZOP      REAL
   2  ZT        REAL               BK7        1510  ZTP      REAL

FILE NAMES       MODE
      INPUT     FMT                OUTPUT   FMT

EXTERNALS        TYPE   ARGS
      DERIVE              0                     READMS            4
      SQRT      REAL      1 LIBRARY

INLINE FUNCTIONS TYPE   ARGS
      FLOAT     REAL      1  INTRIN

STATEMENT LABELS
  35  100                            0  120                             267  130
 341  140                            0  145                             522  240
 532  250                            0  290                             615  300
```

```
  626  310                         636  320                         722  325
    0  330                           0  335                        1146  350
 1307  10000   FMT                1312  10100   FMT                1314  10200   FMT
 1323  10300   FMT                1354  10400   FMT                1376  10500   FMT
 1416  10600   FMT                1422  10700   FMT

       SUBROUTINE OFSIM 74/875 OPT=2 FTN 4.8+670  87/09/01. 13.42.10  PAGE 11
LOOPS  LABEL    INDEX     FROM-TO    LENGTH    PROPERTIES
    7  350      NP          39 398    1142B             EXT REFS  NOT INNER
   27  100      NS          52 57       10B             EXT REFS    EXITS
  101  335      JP          76 397    1045B             EXT REFS  NOT INNER
  115  335      IP          78 397    1027B             EXT REFS  NOT INNER
  122  120      M           85 87        2B     INSTACK
  131  330      NS          91 338     601B             EXT REFS  NOT INNER
  272  325      JS         141 337     435B             EXT REFS  NOT INNER
  300  325      IS         143 337     425B             EXT REFS  NOT INNER
  343  145      LS         173 208     124B             EXT REFS  NOT INNER
  350  145      KS         175 208     115B             EXT REFS
  507  290      NUM        221 283     106B       OPT
COMMON BLOCKS   LENGTH
       BK1       19735
       BK2          18
       BK3          11
       BK4         575
       BK7          23
       CONSTS       81
       IN1          24
       IN2         196
       IN3          12
       IN4           9
       IN4A         11
       IN5          16
STATISTICS
  PROGRAM LENGTH                1633B     923
  SCM LABELED COMMON LENGTH    50347B   20711
```

```
 66300B SCM USED

SUBROUTINE PRCMAT 74/875 OPT=2 FTN 4.8+670  87/09/01. 13.42.10  PAGE 1
      SUBROUTINE PRCMAT(NSEAM,NBXE,NY,NX,MDKOD,NBET)
C
C.. SUBROUTINE TO DISPLAY EITHER COARSE MESH PROPERTY CODES
C   OR COARSE MESH EXTRACTION RATIO CODES
C
      DIMENSION MDKOD(3200), ITMP(50)
C
      PRINT 10200
C
      DO 4000 NP=1, NSEAM
        NO = (NP-1) * NBXE
        PRINT 10260,NP
        DO 3900 IV1 = 1,NY
          IV2 = NY + 1 - IV1
          DO 3800 IV3 = 1,NX
            IDX =  (IV3-1) * NBET + IV2 + NO
            ITMP(IV3) = MDKOD(IDX)
 3800     CONTINUE
          PRINT 10300, (ITMP(I),I=1,NX)
 3900   CONTINUE
 4000 CONTINUE
C
10200 FORMAT(//)
10260 FORMAT(/*  NO. OF SEAM-*,I5/)
10300 FORMAT(10X,50A1)
      RETURN
       END

  SYMBOLIC REFERENCE MAP (R=1)
ENTRY POINTS
   3  PRCMAT
VARIABLES        TYPE          RELOCATION
  75  I          INTEGER                         74  IDX       INTEGER
```

```
   76  ITMP      INTEGER   ARRAY                      71  IV1       INTEGER
   72  IV2       INTEGER                              73  IV3       INTEGER
    0  MDKOD     INTEGER   ARRAY      F.P.             0  NBET      INTEGER              F.P.
    0  NBXE      INTEGER              F.P.            67  NP        INTEGER
    0  NSEAM     INTEGER              F.P.             0  NX        INTEGER              F.P.
    0  NY        INTEGER              F.P.            70  NO        INTEGER
FILE NAMES        MODE
      OUTPUT     FMT
STATEMENT LABELS
    0  3800                                  0  3900                                   0  4000
   57  10200   FMT                          61  10260   FMT                           65  10300   FMT

LOOPS  LABEL    INDEX     FROM-TO    LENGTH    PROPERTIES
   15  4000     NP          10 21       26B              EXT REFS  NOT INNER
   24  3900     IV1         13 20       14B              EXT REFS  NOT INNER
   30  3800     IV3         15 18        2B     INSTACK
```

SUBROUTINE PRCMAT 74/875 OPT=2 FTN 4.8+670 87/09/01. 13.42.10 PAGE 2

```
STATISTICS
  PROGRAM LENGTH                 167B      119
        60000B SCM USED
```

SUBROUTINE PRMAT 74/875 OPT=2 FTN 4.8+670 87/09/01. 13.42.10 PAGE 1

```
      SUBROUTINE PRMAT(NSEAM,IMY,JMX,NBLKX,NBLKY,MCODE)
C
C.. SUBROUTINE TO DISPLAY FINE MESH MATERIAL PROPERTIES OR
C                           FINE MESH MINING CODES
C
      DIMENSION MCODE(2403), INDEX(100), KODE(25)
C

      N4 = JMX * 5
      ILINE = 0
C
      DO 1000 NP=1,NSEAM
```

```
      PRINT 10260, NP
      NO = (NP-1) * NBLKX * NBLKY * 3 + 1
C
      DO 900 IP1=1,IMY
        IP = IMY + 1 - IP1
        N1 = NO + (IP-1) * 3
C
        DO 800 I1=1,5
          I = 5 - I1 + 1
          N = 0
C
          DO 700 JP=1,JMX
            N2 = N1 + (JP-1) * IMY * 3
            DECODE (25, 10300, MCODE(N2)) KODE
C
            DO 600 J=1,5
              N = N + 1
              N3 = (J - 1) * 5 + I
              INDEX(N) = KODE(N3)
  600       CONTINUE
C
  700     CONTINUE
          PRINT 10310, (INDEX(M),M=1,N4)
          ILINE = ILINE + 1
          IF (ILINE .LT. 5) GO TO 800
            ILINE = 0
            PRINT 10320
C
  800   CONTINUE
C
  900 CONTINUE
C
 1000 CONTINUE
C
10260 FORMAT(/*  NO. OF SEAM-*,I5/)
10300 FORMAT(25A1)
```

```
             10310 FORMAT(10X,20(5A1,1X))
             10320 FORMAT(* *)
                   RETURN
                   END

       SUBROUTINE PRMAT 74/875 OPT=2 FTN 4.8+670  87/09/01. 13.42.10  PAGE 2
       SYMBOLIC REFERENCE MAP (R=1)
ENTRY POINTS
    3  PRMAT
VARIABLES          TYPE          RELOCATION
  152  I         INTEGER                        143  ILINE     INTEGER
    0  IMY       INTEGER                F.P.    161  INDEX     INTEGER   ARRAY
  147  IP        INTEGER                        146  IP1       INTEGER
  151  I1        INTEGER                        156  J         INTEGER
    0  JMX       INTEGER                F.P.    154  JP        INTEGER
  325  KODE      INTEGER    ARRAY               160  M         INTEGER
    0  MCODE     INTEGER    ARRAY       F.P.    153  N         INTEGER
    0  NBLKX     INTEGER                F.P.      0  NBLKY     INTEGER             F.P.
  144  NP        INTEGER                          0  NSEAM     INTEGER             F.P.
  145  NO        INTEGER                        150  N1        INTEGER
  155  N2        INTEGER                        157  N3        INTEGER
  142  N4        INTEGER
FILE NAMES         MODE
       OUTPUT    FMT
STATEMENT LABELS
    0  600                                  0  700                                72  800
    0  900                                  0  1000                              126  10260   FMT
  132  10300   FMT                        134  10310   FMT                       137  10320   FMT

LOOPS  LABEL     INDEX       FROM-TO     LENGTH     PROPERTIES
   22  1000      NP           12 45        60B                 EXT REFS  NOT INNER
   30  900       IP1          16 43        47B                 EXT REFS  NOT INNER
   35  800       I1           20 41        40B                 EXT REFS  NOT INNER
   43  700       JP           24 34        20B                 EXT REFS  NOT INNER
   54  600       J            28 32         2B      INSTACK
```

```
STATISTICS
  PROGRAM LENGTH                363B     243
        60000B SCM USED

    SUBROUTINE PRTINP 74/875 OPT=2 FTN 4.8+670  87/09/01. 13.42.10  PAGE 1

                SUBROUTINE PRTINP
          C
          C..THIS SUBROUTINE WILL PRINT OUT ALL INPUT DATA READ
          C  BY SUBROUTINE RDEDIT.
          C
                COMMON/BK1/    DUB(3200), DVB(3200), DWB(3200),DWBI(3200),
               #               DE(2100),  INDEX(805),  KODE(25), INDX13(805)
               #               ,NGOB(3200)
                COMMON/BK3/    EN(3,3), IMAX, JMAX
                COMMON/BK4/    S13U(3300), S23U(3300), S33U(3300),
               #               S23V(3300), S33V(3300), S33W(3300),
               #               CF(3,27,4)
                COMMON/CONSTS/ PI, FACT, V1, V2, CONS, COND, CON, IAMAT(28),
               #               HEW, HBW, EW, RLIM, NBXE, NBT, KB, NTR, ITMINI,
               #               ITMAXI, IPHASE, JOG, ID(4,4), LQR, KI, KE, NCUT,
               #               ACC, NONE, KD0, KD1, KD2, KD3, KD4, KD5, KD6
               #               , NELMX, NELMY, NBLKX, NBLKY, NBLKFM
                COMMON/IN1/    TITLE(20), V, E, NSEAM, NMATS
                COMMON/IN2/    ES(28), GS(28), COHES(28),
               #               PHI(28), COHES1(28), PHI1(28), IGOB(28)
                COMMON/IN3/    A11, A12, A13, A22, A23, A33,
               #               B11, B12, B13, B22, B23, B33
                COMMON/IN4/    BW, NBXI, NBET, NSYM, NOSP, IFXS, IFXE, IFYS, IFYE
                COMMON/IN4A/   IEX, EXR(10)
                COMMON/IN5/    XO(4), YO(4), ZO(4), THIKNS(4)
                COMMON/IN6/    STRANG, DIPANG, OFSANG
                COMMON/IN7/    ORF, ITMAX, NRUN, ITP, NGRID
                COMMON/IN9/    MKOD(2403), MCKOD(3200), MCEKOD(3200)
                DIMENSION MATKOD(2403)
                EQUIVALENCE (MATKOD,S13U)
```

```
      DIMENSION IDIGIT(9)
      DATA IDIGIT /1H1,1H2,1H3,1H4,1H5,1H6,1H7,1H8,1H9/
C
      PRINT 10000, TITLE
C
C
C..THE MATERIAL PROPERTIES
C
      PRINT 10010
      PRINT 10020,V,E,NSEAM,NMATS
      NMATS2 = NMATS + 2
      DO 10 N=1,NMATS
        PRINT 10030,N
        IF(IGOB(N) .EQ. 1) PRINT 10095
        PRINT 10040,ES(N)
        PRINT 10050,GS(N)
        PRINT 10060,COHES(N)
        PRINT 10070,PHI(N)
        PRINT 10080,COHES1(N)
        PRINT 10090,PHI1(N)
   10 CONTINUE
C
C  THE PRIMITIVE STRESS PARAMETERS
C
      PRINT 10100
      PRINT 10110,A11,B11,A12,B12,A13,B13,A22,B22,A23,B23,A33,B33
C
```

SUBROUTINE PRTINP 74/875 OPT=2 FTN 4.8+670 87/09/01. 13.42.10 PAGE 2

```
C  MODEL CONTROL AND MODEL DATA
C
C
      PRINT 10120
      PRINT 10130,NSEAM,BW,NBXI,NBET,NSYM,NOSP,IFXS,IFXE,IFYS,IFYE
      PRINT 10135,IEX,(EXR(I),I=1,IEX)
```

```
C     PRINT 10140, STRANG,DIPANG,OFSANG
      PRINT 10150
      PRINT 10160,(N,XO(N),YO(N),ZO(N),THIKNS(N),N=1,NSEAM)
      PRINT 10170
      PRINT 10180,((EN(I,J),I=1,3),J=1,3)
C
C  PROGRAM FLOW PARAMETERS
C
      PRINT 10190
      PRINT 10200, ORF, ITMAX
      IF (NRUN. GT. 1) GO TO 100
        PRINT 10210
      GO TO 200
  100   PRINT 10220, NRUN, ITP
C
  200 IF (NGRID .GT. 0) GO TO 300
        PRINT 10240
      GO TO 400
  300   PRINT 10230
C..DISPLAY THE MATERIAL PROPERTIES CODES FOR THE FINE MESH BLOCKS
C
  400 PRINT 10290,(IAMAT(I),I=1,NMATS2)
      I=IFYE-IFYS+1
      J=IFXE-IFXS+1
      CALL PRMAT(NSEAM,I,J,NBLKX,NBLKY,MATKOD)
C
C..DISPLAY THE MINING CODES FOR THE FINE MESH BLOCKS
C
      PRINT 10250
      CALL PRMAT(NSEAM,I,J,NBLKX,NBLKY,MKOD)
C
C..CHECK TO SEE IF COARSE MESH EXISTS
C
      IF( NBXI .EQ. IFXE-IFXS+1 .AND. NBET .EQ. IFYE-IFYS+1) GO TO 2000
C
C.. DISPLAY MATERIAL PROPERTIES FOR COARSE MESH BLOCKS
```

```
      C
            PRINT 10400,(IAMAT(I),I=1,NMATS2)
             CALL PRCMAT(NSEAM,NBXE,IMAX,JMAX,MCKOD,NBET)
      C
      C..CONVERT MATERIAL PROPERTIES TO ARITHMETIC VALUES
      C
                  DO 660 J = 1,NBT
      C
      C..SKIP IF FINE MESH BLOCK
      C
                    IF( MCKOD(J) .EQ. 1H* ) GO TO 660
      C
                    I = 1
        610         IF (MCKOD(J).EQ.IAMAT(I)) GOTO 620
                    I = I + 1
```

SUBROUTINE PRTINP 74/875 OPT=2 FTN 4.8+670 87/09/01. 13.42.10 PAGE 3

```
                    IF (I.LE.NMATS2) GOTO 610
      C
      C   ERROR STOP
      C
                    PRINT 10700,J,MCKOD(J),(IAMAT(I),I=1,NMATS2)
                    STOP
      C
      C
        620         IF (I-2) 630,640,650
        630         MCKOD(J) = 27
                    GOTO 660
        640         MCKOD(J) = 28
                    GOTO 660
        650         MCKOD(J) = I - 2
        660       CONTINUE
      C
      C
      C.. DISPLAY EXTRACTION RATIO CODES FOR COARSE MESH BLOCKS
```

```
C
        PRINT 10600
        PRINT 10135, IEX, (EXR(I),I=1,IEX)
C
        CALL PRCMAT(NSEAM,NBXE,IMAX,JMAX,MCEKOD,NBET)
C
C
C.. CONVERT EXTRACTION RATIO CODES FOR INDEXING INTO EXTR RATIO ARRAY
C
      DO 1000 I=1,NBT
C
        DO 900 J=1,9
          IF(MCEKOD(I) .NE. IDIGIT(J)) GO TO 900
          MCEKOD(I) = J
          GO TO 1000
  900   CONTINUE
C
        MCEKOD(I) = 10
 1000 CONTINUE
C
C..OUTPUT FORMATS
C
10000 FORMAT (1H1, 20A4)
10010 FORMAT (/*  IN THE FOLLOWING X,Y,Z REFER TO GLOBAL AXES OR COORDIN
     1ATES*/*  AND X1,X2,X3  OR  1,2,3 REFER TO THE LOCAL AXES OR COORDI
     2NATES.*//*  MATERIAL PROPERTIES   --*)
10020 FORMAT(/*  POISSON"S RATIO OF ROCK MASS          --*,F15.2,
     1       //*  MODULUS OF ELASTICITY OF ROCK MASS    --*,F15.2,
     2       //*  NO. OF SEAMS*,28X,I5,
     3       //*  NO. OF MATERIALS*,24X,I5)
10030 FORMAT (/* MATERIAL PROPERTY SET NUMBER --*, I3)
10040 FORMAT( /*  MODULUS OF ELASTICITY OF SEAM MATERIAL--*,4F15.2)
10050 FORMAT (/*  MODULUS OF RIGIDITY OF SEAM MATERIAL  --*,4F15.2)
10060 FORMAT (/*  COHESION OF THE SEAM MATERIAL         --*,4F15.2)
10070 FORMAT (/*  FRICTION ANGLE OF THE SEAM MATERIAL   --*,4F15.2)
10080 FORMAT (/*  RESIDUAL VALUE OF THE COHESION        --*,4F15.2)
```

```
      10090 FORMAT (/*  RESIDUAL VALUE OF THE FRICTION ANGLE  --*,4F15.2)
      10095 FORMAT (//*   THIS IS A GOB OR INSERTED MATERIAL --*)
      10100 FORMAT(//*  PRIMITIVE STRESS PARAMETERS  --*)
SUBROUTINE PRTINP 74/875 OPT=2 FTN 4.8+670  87/09/01. 13.42.10  PAGE 4

      10110 FORMAT(/*  PRIMITIVE STRESSES ARE GIVEN AS--* /*  PXX=*,F8.4,*--*,
           1F8.4,* Z*/*  PXY=*,F8.4,*--*,F8.4,* Z*/*  PXZ=*,F8.4,*--*,F8.4,* Z
           2*/*  PYY=*,F8.4,*--*,F8.4,* Z*/*  PYZ=*,F8.4,*--*,F8.4,* Z*/*  PZZ
           3=*,F8.4,*--*,F8.4,* Z*/)
      10120 FORMAT(//*  MODEL CONTROL AND MODEL DATA  --*)
      10130 FORMAT( /*  NO. OF PARALLEL SEAMS                     --*,I8//
           1         *  WIDTH OF BLOCKS                           --*,F8.2//
           2         *  NO. OF BLOCKS ALONG X1 AXIS               --*,I8//
           3         *  NO. OF BLOCKS ALONG X2 AXIS               --*,I8//
           4         *  SYMMETRY CODE SPECIFIED                   --*,I8//
           5         *  NUMBER OF OFF-SEAM PLANES SELECTED        --*,I8//
           6         *  FINE MESH STARTING BLOCK X-AXIS           --*,I8//
           7         *  FINE MESH ENDING   BLOCK X-AXIS           --*,I8//
           8         *  FINE MESH STARTING BLOCK Y-AXIS           --*,I8//
           9         *  FINE MESH ENDING   BLOCK Y-AXIS           --*,I8//)
      10135 FORMAT( /*  NUMBER OF EXTRACTION CODES USED IN MODEL --*,I8/
           1        /*  EXTRACTION RATIO CODE TABLE : *,9F10.3//)
      10140 FORMAT (/* STRIKE ANGLE--*,F15.2,/
           #         *    DIP ANGLE--*,F15.2,/
           #         * OFFSET ANGLE--*,F15.2)
      10150 FORMAT(/*  GLOBAL COORDINATES OF THE LOCAL ORIGINS AND SEAM THICKN
           1ESSES --*
           2  /*  SEAM NO.         X            Y            Z     THICKNESS*)
      10160 FORMAT(I8,4F12.2)
      10170 FORMAT(/*  ORIENTATION OF THE SEAMS- DIRECTION COSINES OF THE LOCA
           1L AXES */*  WITH RESPECT TO THE GLOBAL AXES --*//
           2 *     EN(1,X)     EN(2,X)     EN(3,X)     EN(1,Y)     EN(2,Y)
           3 EN(3,Y)     EN(1,Z)     EN(2,Z)     EN(3,Z)*)
      10180 FORMAT(9F12.5)
      10190 FORMAT(//*  PROGRAM FLOW CONTROL PARAMETERS  --*)
      10200 FORMAT( /*  OVER RELAXATION FACTOR                    --*,F8.2//
```

```
      1        *  MAXIMUM NO. OF ITERATIONS SPECIFIED  --*,I8)
10210 FORMAT( /*  THIS IS THE MAIDEN COMPUTER RUN FOR THIS PROBLEM *)
10220 FORMAT( /*  THIS IS COMPUTER RUN NO.*,I3,* FOR THIS PROBLEM,*,
      1   I4,* ITERATION CYCLES WERE COMPLETED IN PREVIOUS RUNS*)
10230 FORMAT(*  GRID DETAILS ARE SAME AS IN THE PREVIOUS COMPUTER RUN, I
      1NFLUENCE COEFFICIENTS ARE RETRIEVED FROM STORAGE AREA*)
10240 FORMAT( /*  THIS IS A NEW GRID, INFLUENCE COEFFICIENTS ARE COMPUTE
      1D WITHIN THIS COMPUTER RUN*)
10250 FORMAT(/*  MINING KODES        -   MINED OUT (1) */
      1        *                         UNMINED, SEAM MATERIAL -*/
      2        *                         RIGID (2)*/
      3        *                         ELASTIC (3)*/
      4        *                         MOHR COULOMB TYPE (4)*/
      5        *                         MOHR COULOMB TYPE-YIELDED (5)*
      6       /*                         MOHR COULOMB TYPE-CRACKED (6)*
      7       /*                         MOHR COULOMB TYPE-CRACKED AND
      8CLOSED BACK (7)*/
      9        *                         MOHR COULOMB TYPE-CRACKED OPEN
      9 CLOSED BACK AND YIELDED (8)*)
10260 FORMAT(/*  NO. OF SEAM-*,I5/)
10270 FORMAT(25I1)
10280 FORMAT (10X,20(5I1,1X))
10290 FORMAT (*1*//* MINE SEAM PROPERTIES:*//
      #*     MINED OUT = *,A1//
      #*         RIGID = *,A1//
      #* SEAM PROPERTIES ARE:*//28(1X,A1))
```

SUBROUTINE PRTINP 74/875 OPT=2 FTN 4.8+670 87/09/01. 13.42.10 PAGE 5

```
10300 FORMAT (25A1)
10310 FORMAT (10X,20(5A1,1X))
10320 FORMAT (* *)
C
10400 FORMAT (*1*//* MINE SEAM PROPERTIES: FOR COARSE MESH *//
      #*     MINED OUT = *,A1//
      #*         RIGID = *,A1//
      #* SEAM PROPERTIES ARE:*//28(1X,A1))
10600 FORMAT(/*     EXTRACTION RATIO CODES FOR COARSE MESH */)
```

```
          10700 FORMAT(//* --- ERROR:  BAD MATERIAL PROPERTY CODE ---*/
                #/* BLOCK  = *,I3,/
  240           #                  * PROPERTY CODE IS: *,A1/
                #/* VALID CODES ARE:*/28(1X,A1))
           2000 RETURN
                END
    SYMBOLIC REFERENCE MAP (R=1)
ENTRY POINTS
   1  PRTINP
VARIABLES       TYPE          RELOCATION
  103  ACC       REAL               CONSTS           0  A11       REAL               IN3
    1  A12       REAL               IN3              2  A13       REAL               IN3
    3  A22       REAL               IN3              4  A23       REAL               IN3
    5  A33       REAL               IN3              0  BW        REAL               IN4
    6  B11       REAL               IN3              7  B12       REAL               IN3
   10  B13       REAL               IN3             11  B22       REAL               IN3
   12  B23       REAL               IN3             13  B33       REAL               IN3
46530  CF        REAL      ARRAY    BK4             70  COHES     REAL      ARRAY    IN2
  160  COHES1    REAL      ARRAY    IN2              6  CON       REAL               CONSTS
    5  COND      REAL               CONSTS           4  CONS      REAL               CONSTS
31000  DE        REAL      ARRAY    BK1              1  DIPANG    REAL               IN6
    0  DUB       REAL      ARRAY    BK1           6200  DVB       REAL      ARRAY    BK1
14400  DWB       REAL      ARRAY    BK1          22600  DWBI      REAL      ARRAY    BK1
   25  E         REAL               IN1              0  EN        REAL      ARRAY    BK3
    0  ES        REAL      ARRAY    IN2             45  EW        REAL               CONSTS
    1  EXR       REAL      ARRAY    IN4A             1  FACT      REAL               CONSTS
   34  GS        REAL      ARRAY    IN2             44  HBW       REAL               CONSTS
   43  HEW       REAL               CONSTS        1423  I         INTEGER
    7  IAMAT     INTEGER   ARRAY    CONSTS          57  ID        INTEGER   ARRAY    CONSTS
 1425  IDIGIT    INTEGER   ARRAY                     0  IEX       INTEGER            IN4A
    6  IFXE      INTEGER            IN4              5  IFXS      INTEGER            IN4
   10  IFYE      INTEGER            IN4              7  IFYS      INTEGER            IN4
  250  IGOB      INTEGER   ARRAY    IN2             11  IMAX      INTEGER            BK3
35064  INDEX     INTEGER   ARRAY    BK1          36562  INDX13    INTEGER   ARRAY    BK1
   55  IPHASE    INTEGER            CONSTS           1  ITMAX     INTEGER            IN7
   54  ITMAXI    INTEGER            CONSTS          53  ITMINI    INTEGER            CONSTS
```

```
     3  ITP       INTEGER             IN7         1424  J         INTEGER
    12  JMAX      INTEGER             BK3           56  JOG       INTEGER           CONSTS
    51  KB        INTEGER             CONSTS       105  KD0       INTEGER           CONSTS
   106  KD1       INTEGER             CONSTS       107' KD2       INTEGER           CONSTS
   110  KD3       INTEGER             CONSTS       111  KD4       INTEGER           CONSTS

        SUBROUTINE PRTINP 74/875 OPT=2 FTN 4.8+670  87/09/01. 13.42.10  PAGE 6
VARIABLES          TYPE           RELOCATION
   112  KD5       INTEGER             CONSTS       113  KD6       INTEGER           CONSTS
   101  KE        INTEGER             CONSTS       100  KI        INTEGER           CONSTS
 36531  KODE      INTEGER   ARRAY     BK1           77  LQR       INTEGER           CONSTS
     0  MATKOD    INTEGER   ARRAY     BK4        12743  MCEKOD    INTEGER   ARRAY   IN9
  4543  MCKOD     INTEGER   ARRAY     IN9            0  MKOD      INTEGER   ARRAY   IN9
  1422  N         INTEGER                            2  NBET      INTEGER           IN4
   120  NBLKFM    INTEGER             CONSTS       116  NBLKX     INTEGER           CONSTS
   117  NBLKY     INTEGER             CONSTS        50  NBT       INTEGER           CONSTS
    47  NBXE      INTEGER             CONSTS         1  NBXI      INTEGER           IN4
   102  NCUT      INTEGER             CONSTS       114  NELMX     INTEGER           CONSTS
   115  NELMY     INTEGER             CONSTS     40227  NGOB      INTEGER   ARRAY   BK1
     4  NGRID     INTEGER             IN7           27  NMATS     INTEGER           IN1
  1421  NMATS2    INTEGER                          104  NONE      INTEGER           CONSTS
     4  NOSP      INTEGER             IN4            2  NRUN      INTEGER           IN7
    26  NSEAM     INTEGER             IN1            3  NSYM      INTEGER           IN4
    52  NTR       INTEGER             CONSTS         2  OFSANG    REAL              IN6
     0  ORF       REAL                IN7          124  PHI       REAL      ARRAY   IN2
   214  PHI1      REAL      ARRAY     IN2            0  PI        REAL              CONSTS
    46  RLIM      REAL                CONSTS         0  STRANG    REAL              IN6
     0  S13U      REAL      ARRAY     BK4         6344  S23U      REAL      ARRAY   BK4
 23254  S23V      REAL      ARRAY     BK4        14710  S33U      REAL      ARRAY   BK4
 31620  S33V      REAL      ARRAY     BK4        40164  S33W      REAL      ARRAY   BK4
    14  THIKNS    REAL      ARRAY     IN5            0  TITLE     REAL      ARRAY   IN1
    24  V         REAL                IN1            2  V1        REAL              CONSTS
     3  V2        REAL                CONSTS         0  XO        REAL      ARRAY   IN5
     4  YO        REAL      ARRAY     IN5           10  ZO        REAL      ARRAY   IN5

FILE NAMES         MODE
```

```
         OUTPUT     FMT

EXTERNALS            TYPE    ARGS
         PRCMAT                 6                                          PRMAT                 6

STATEMENT LABELS
     0   10                                           132  100                                      134  200
   140   300                                          142  400                                      210  610
   223   620                                            0  630          INACTIVE                    227  640
   231   650                                          233  660                                      255  900
   257   1000                                         260  2000                                     545  10000  FMT
   547   10010  FMT                                   571  10020  FMT                               613  10030  FMT
   621   10040  FMT                                   630  10050  FMT                               637  10060  FMT
   646   10070  FMT                                   655  10080  FMT                               664  10090  FMT
   673   10095  FMT                                   701  10100  FMT                               707  10110  FMT
   741   10120  FMT                                   747  10130  FMT                              1032  10135  FMT
  1045   10140  FMT  NO REFS                         1056  10150  FMT                              1075  10160  FMT
  1100   10170  FMT                                  1130  10180  FMT                              1132  10190  FMT
  1140   10200  FMT                                  1154  10210  FMT                              1163  10220  FMT
  1200   10230  FMT                                  1216  10240  FMT                              1230  10250  FMT
  1315   10260  FMT  NO REFS                         1321  10270  FMT  NO REFS                     1323  10280  FMT  NO REFS
  1326   10290  FMT                                  1344  10300  FMT  NO REFS                     1346  10310  FMT  NO REFS
  1351   10320  FMT  NO REFS                         1353  10400  FMT                              1372  10600  FMT
  1401   10700  FMT

LOOPS   LABEL     INDEX      FROM-TO     LENGTH     PROPERTIES
   15   10        N             42 51       41B                EXT REFS
  102             N             66 66       12B                EXT REFS
  205   660       J            106 129      27B                EXT REFS

        SUBROUTINE PRTINP 74/875 OPT=2 FTN 4.8+670  87/09/01. 13.42.10  PAGE 7
LOOPS   LABEL     INDEX      FROM-TO     LENGTH     PROPERTIES
  251   1000      I            142 151       7B                NOT INNER
  253   900       J            144 148       3B      INSTACK    EXITS

COMMON BLOCKS    LENGTH
```

```
      BK1         19735
      BK3            11
      BK4         20124
      CONSTS         81
      IN1            24
      IN2           196
      IN3            12
      IN4             9
      IN4A           11
      IN5            16
      IN6             3
      IN7             5
      IN9          8803

STATISTICS
  PROGRAM LENGTH                   1436B     798
  SCM LABELED COMMON LENGTH  137606B   49030
        60000B SCM USED
```

SUBROUTINE PSTRES 74/855 OPT=2 FTN 4.8+587 86/02/05. 13.56.15 PAGE 1

```
      SUBROUTINE PSTRES
C
C THIS ROUTINE OBTAINS THE NORMAL AND SHEAR STRESS COMPONENTS (PO1(N),
C PO2(N),PO3(N)) ACROSS EACH SEAM (N) AT THEIR LOCAL ORIGIN AND ALSO
C COMPUTES THE CHANGE IN THESE COMPONENTS (P1XI,P2ET,P3XI..ETC) OVER
C HALF WIDTH OF ELEMENTS.
C
      COMMON/BK2/    PO1(4), PO2(4), PO3(4),
     #               P1XI,   P2XI,   P3XI,
     #               P1ET,   P2ET,   P3ET
      COMMON/BK3/    EN(3,3), IMAX, JMAX
      COMMON/CONSTS/ PI, FACT, V1, V2, CONS, COND, CON, IAMAT(28),
     #               HEW, HBW, EW, RLIM, NBXE, NBT, KB, NTR, ITMINI,
     #               ITMAXI, IPHASE, JOG, ID(4,4), LQR, KI, KE, NCUT,
     #               ACC, NONE, KD0, KD1, KD2, KD3, KD4, KD5, KD6
```

```
     #                 , NELMX, NELMY, NBLKX, NBLKY, NBLKFM
      COMMON/IN1/    TITLE(20), V, E, NSEAM, NMATS
      COMMON/IN3/    A11, A12, A13, A22, A23, A33,
     #               B11, B12, B13, B22, B23, B33
      COMMON/IN5/    XO(4), YO(4), ZO(4), THIKNS(4)
C
      DIMENSION AP(6), BP(6), P(3,3)
      EQUIVALENCE (A11,AP),(B11,BP)
C
      DATA P,PO1,PO2,PO3,P1XI,P2XI,P3XI,P1ET,P2ET,P3ET/27*0.0/
C
C  STRESSES AT THE LOCAL ORIGIN--
C
      DO 85 N=1,NSEAM
        P(1,1)=AP(1)-BP(1)*ZO(N)
        P(1,2)=AP(2)-BP(2)*ZO(N)
        P(1,3)=AP(3)-BP(3)*ZO(N)
        P(2,2)=AP(4)-BP(4)*ZO(N)
        P(2,3)=AP(5)-BP(5)*ZO(N)
        P(3,3)=AP(6)-BP(6)*ZO(N)
        P(2,1)=P(1,2)
        P(3,1)=P(1,3)
        P(3,2)=P(2,3)
C
        DO 80 I=1,3
C
          DO 80 J=1,3
            PO1(N)=PO1(N)+EN(1,I)*EN(3,J)*P(I,J)
            PO2(N)=PO2(N)+EN(2,I)*EN(3,J)*P(I,J)
   80       PO3(N)=PO3(N)+EN(3,I)*EN(3,J)*P(I,J)
   85 CONTINUE
C
C  STRESS GRADIENTS ALONG X1 AXIS--
C
      P(1,1)=BP(1)*HEW*EN(1,3)
      P(1,2)=BP(2)*HEW*EN(1,3)
```

```
      P(1,3)=BP(3)*HEW*EN(1,3)
      P(2,2)=BP(4)*HEW*EN(1,3)
      P(2,3)=BP(5)*HEW*EN(1,3)
      P(3,3)=BP(6)*HEW*EN(1,3)
      P(2,1)=P(1,2)
      P(3,1)=P(1,3)

SUBROUTINE PSTRES 74/855 OPT=2 FTN 4.8+587  86/02/05. 13.56.15  PAGE 2
      P(3,2)=P(2,3)
C
      DO 95 I=1,3
C
        DO 95 J=1,3
          P1XI=P1XI+EN(1,I)*EN(3,J)*P(I,J)
          P2XI=P2XI+EN(2,I)*EN(3,J)*P(I,J)
   95     P3XI=P3XI+EN(3,I)*EN(3,J)*P(I,J)
C
C  STRESS GRADIENTS ALONG X2 AXIS--
C
      P(1,1)=BP(1)*HEW*EN(2,3)
      P(1,2)=BP(2)*HEW*EN(2,3)
      P(1,3)=BP(3)*HEW*EN(2,3)
      P(2,2)=BP(4)*HEW*EN(2,3)
      P(2,3)=BP(5)*HEW*EN(2,3)
      P(3,3)=BP(6)*HEW*EN(2,3)
      P(2,1)=P(1,2)
      P(3,1)=P(1,3)
      P(3,2)=P(2,3)
C
      DO 105 I=1,3
C
        DO 105 J=1,3
          P1ET=P1ET+EN(1,I)*EN(3,J)*P(I,J)
          P2ET=P2ET+EN(2,I)*EN(3,J)*P(I,J)
  105     P3ET=P3ET+EN(3,I)*EN(3,J)*P(I,J)
C
```

```
                    RETURN
                    END
     SYMBOLIC REFERENCE MAP (R=1)
ENTRY POINTS
   1  PSTRES
VARIABLES         TYPE            RELOCATION
  103  ACC        REAL                CONSTS        0  AP        REAL       ARRAY    IN3
    0  A11        REAL                IN3           1  A12       REAL                IN3
    2  A13        REAL                IN3           3  A22       REAL                IN3
    4  A23        REAL                IN3           5  A33       REAL                IN3
    6  BP         REAL       ARRAY    IN3           6  B11       REAL                IN3
    7  B12        REAL                IN3          10  B13       REAL                IN3
   11  B22        REAL                IN3          12  B23       REAL                IN3
   13  B33        REAL                IN3           6  CON       REAL                CONSTS
    5  COND       REAL                CONSTS        4  CONS      REAL                CONSTS
   25  E          REAL                IN1           0  EN        REAL       ARRAY    BK3
   45  EW         REAL                CONSTS        1  FACT      REAL                CONSTS
   44  HBW        REAL                CONSTS       43  HEW       REAL                CONSTS
  152  I          INTEGER                           7  IAMAT     INTEGER    ARRAY    CONSTS
   57  ID         INTEGER    ARRAY    CONSTS       11  IMAX      INTEGER             BK3
   55  IPHASE     INTEGER             CONSTS       54  ITMAXI    INTEGER             CONSTS
   53  ITMINI     INTEGER             CONSTS      153  J         INTEGER

     SUBROUTINE PSTRES 74/855 OPT=2 FTN 4.8+587  86/02/05. 13.56.15  PAGE 3
VARIABLES         TYPE            RELOCATION
   12  JMAX       INTEGER             BK3          56  JOG       INTEGER             CONSTS
   51  KB         INTEGER             CONSTS      105  KD0       INTEGER             CONSTS
  106  KD1        INTEGER             CONSTS      107  KD2       INTEGER             CONSTS
  110  KD3        INTEGER             CONSTS      111  KD4       INTEGER             CONSTS
  112  KD5        INTEGER             CONSTS      113  KD6       INTEGER             CONSTS
  101  KE         INTEGER             CONSTS      100  KI        INTEGER             CONSTS
   77  LQR        INTEGER             CONSTS      151  N         INTEGER
  120  NBLKFM     INTEGER             CONSTS      116  NBLKX     INTEGER             CONSTS
  117  NBLKY      INTEGER             CONSTS       50  NBT       INTEGER             CONSTS
   47  NBXE       INTEGER             CONSTS      102  NCUT      INTEGER             CONSTS
  114  NELMX      INTEGER             CONSTS      115  NELMY     INTEGER             CONSTS
```

```
   27  NMATS     INTEGER           IN1            104  NONE     INTEGER          CONSTS
   26  NSEAM     INTEGER           IN1             52  NTR      INTEGER          CONSTS
  154  P         REAL      ARRAY                    0  PI       REAL             CONSTS
    0  PO1       REAL      ARRAY   BK2              4  PO2      REAL     ARRAY   BK2
   10  PO3       REAL      ARRAY   BK2             17  P1ET     REAL             BK2
   14  P1XI      REAL              BK2             20  P2ET     REAL             BK2
   15  P2XI      REAL              BK2             21  P3ET     REAL             BK2
   16  P3XI      REAL              BK2             46  RLIM     REAL             CONSTS
   14  THIKNS    REAL      ARRAY   IN5              0  TITLE    REAL     ARRAY   IN1
   24  V         REAL              IN1              2  V1       REAL             CONSTS
    3  V2        REAL              CONSTS           0  XO       REAL     ARRAY   IN5
    4  YO        REAL      ARRAY   IN5             10  ZO       REAL     ARRAY   IN5
STATEMENT LABELS
    0  80                                 0  85                                 0  95
    0  105
LOOPS  LABEL     INDEX    FROM-TO   LENGTH    PROPERTIES
    7  85        N          29 46      44B              NOT INNER
   32  80        I          40 45      17B              NOT INNER
   37  80        J          42 45       6B        OPT
   72  95        I          60 65      16B              NOT INNER
  100  95        J          62 65       6B        OPT
  130  105       I          79 84      16B              NOT INNER
  136  105       J          81 84       6B        OPT
COMMON BLOCKS    LENGTH
       BK2           18
       BK3           11
       CONSTS        81
       IN1           24
       IN3           12
       IN5           16
STATISTICS
  PROGRAM LENGTH                165B      117
  CM LABELED COMMON LENGTH      242B      162
          60000B CM USED
```

SUBROUTINE RDEDIT 74/875 OPT=2 FTN 4.8+670 87/09/01. 13.42.10 PAGE 1

```
      SUBROUTINE RDEDIT(MANY)
C
C..THIS SUBROUTINE WILL READ AND EDIT INPUT FOR MULSIM
C
      COMMON/BK1/    DUB(3200), DVB(3200), DWB(3200),DWBI(3200),
     #               DE(2100),  INDEX(805),  KODE(25), INDX13(805)
     #               ,NGOB(3200)
      COMMON/BK3/    EN(3,3), IMAX, JMAX
      COMMON/BK4/    S13U(3300), S23U(3300), S33U(3300),
     #               S23V(3300), S33V(3300), S33W(3300),
     #               CF(3,27,4)
      COMMON/CONSTS/ PI, FACT, V1, V2, CONS, COND, CON, IAMAT(28),
     #               HEW, HBW, EW, RLIM, NBXE, NBT, KB, NTR, ITMINI,
     #               ITMAXI, IPHASE, JOG, ID(4,4), LQR, KI, KE, NCUT,
     #               ACC, NONE, KD0, KD1, KD2, KD3, KD4, KD5, KD6
     #               , NELMX, NELMY, NBLKX, NBLKY, NBLKFM
      COMMON/IN1/    TITLE(20), V, E, NSEAM, NMATS
      COMMON/IN2/    ES(28), GS(28), COHES(28),
     #               PHI(28), COHES1(28), PHI1(28), IGOB(28)
      COMMON/IN3/    A11, A12, A13, A22, A23, A33,
     #               B11, B12, B13, B22, B23, B33
      COMMON/IN4/    BW, NBXI, NBET, NSYM, NOSP, IFXS, IFXE, IFYS, IFYE
      COMMON/IN4A/   IEX, EXR(10)
      COMMON/IN5/    XO(4), YO(4), ZO(4), THIKNS(4)
      COMMON/IN6/    STRANG, DIPANG, OFSANG
      COMMON/IN7/    ORF, ITMAX, NRUN, ITP, NGRID
      COMMON/IN9/    MKOD(2403), MCKOD(3200), MCEKOD(3200)
C
      DIMENSION MAT(25), MATKOD(2403), ITMP(5,100), ITMP1(50)
      EQUIVALENCE (MATKOD, S13U)
C
C..TITLE OF RUN
C
      READ 10000, TITLE
C
C..MATERIAL PROPERTIES FOR THE HOST MATERIAL
```

```
C  POISSONS RATIO OF ROCK MASS, MODULUS OF ELASTICITY OF ROCK MASS,
C  NUMBER OF SEAMS, NUMBER OF MATERIALS IN SEAM
C
      READ 10010, V, E, NSEAM
      READ *, NMATS
      NMATS2 = NMATS + 2
C
C..EDIT NSEAM
C
      IF (NSEAM .GT. 0 .AND. NSEAM .LT. 3) GO TO 100
        PRINT 10200
        STOP
C
C..EDIT NUMBER OF MATERIALS
C
  100 IF (NMATS .GT. 0 .AND. NMATS .LT. 29) GO TO 200
        PRINT 10210
        STOP
C
C..MODULUS OF ELASTICITY FOR MATERIAL N,
C  MODULUS OF RIGIDITY   FOR MATERIAL N,
```



```
C  COHESION              FOR MATERIAL N,
C  FRICTION ANGLE        FOR MATERIAL N,
C  RESIDUAL VALUE OF THE COHESION FOR MATERIAL N,
C  RESIDUAL VALUE OF THE FRICTION ANGLE FOR MATERIAL N
C
  200 READ 10020, (ES(N), GS(N), COHES(N), PHI(N), COHES1(N), PHI1(N),
     1   IGOB(N),  N = 1, NMATS)
C
C..PRIMITIVE STRESS PARAMETERS
C
      READ 10030, A11, B11, A12, B12, A13, B13, A22, B22, A23, B23,
     1            A33, B33
C
```

```
C..BLOCK WIDTH, NUMBER OF BLOCKS IN THE X1 DIRECTION,
C               NUMBER OF BLOCKS IN THE X2 DIRECTION,
C               SYMMETRY FLAG,
C               OFF-SEAM CALCULATION FLAG
C
      READ 10040, BW, NBXI, NBET, NSYM, NOSP, IFXS, IFXE, IFYS, IFYE
      IF( NSYM .EQ. 2 .OR. NSYM .EQ. 4 ) NBXI = NBXI * 2
      IF( NSYM .EQ. 3 .OR. NSYM .EQ. 4 ) NBET = NBET * 2
C.. CALCULATE THE NUMBER OF ELEMENTS IN THE FINE MESH X-AXIS
      NELMX = (IFXE - IFXS + 1) * 5
C.. CALCULATE THE NUMBER OF ELEMENTS IN THE FINE MESH Y-AXIS
      NELMY = (IFYE - IFYS + 1) * 5
C.. CALCULATE THE NUMBER OF FINE MESH BLOCKS
      NBLKX = IFXE - IFXS + 1
      NBLKY = IFYE - IFYS + 1
      IF (NSYM.EQ.2.OR.NSYM.EQ.4) NBLKX = NBLKX*2
      IF (NSYM.EQ.3.OR.NSYM.EQ.4) NBLKY = NBLKY*2
C.. TOTAL NUMBER OF BLOCKS IN THE FINE MESH AREA
C
      NBLKFM = NBLKX * NBLKY
      NBTFM = NBLKFM * NSEAM
C
C..CALCULATE STORAGE NEEDED FOR FINE MESH MATERIAL PROPERTIES
C
      NFMST = NBLKFM * 3
C
C..CALCULATE NUMBER OF BLOCKS IN ONE SEAM
C  AND NUMBER OF BLOCKS FOR ALL SEAMS
C
      NBXE = NBXI * NBET
      NBT  = NBXE * NSEAM
C
C
C
C..EDIT NBXI, NBET
C
```

```
      IF (NBXI .LT. 41 .AND. NBET .LT. 41) GO TO 300
        PRINT 10220
        STOP
C.
C..EDIT NSYM
C
  300 IF (NSYM .GT. 0 .AND. NSYM .LT. 5) GO TO 400
        PRINT 10230
SUBROUTINE RDEDIT 74/875 OPT=2 FTN 4.8+670  87/09/01. 13.42.10  PAGE 3
        STOP
C
C.. READ IN NUMBER OF EXTRACTION RATIOS AND EXT RATIO ARRAY
C
  400 READ 10045, IEX, (EXR(I), I=1,9)
      EXR(10) = 0.0
C
C..COORDINATES AT ORIGIN FOR SEAM N
C  THICKNESS              FOR SEAM N
C
      READ 10050, (XO(N), YO(N), ZO(N), THIKNS(N), N = 1, NSEAM)
C
C.. READ IN DIRECTION COSINES
C
      READ 10060,((EN(I,J),I=1,3),J=1,3)
C
C..PROGRAM FLOW PARAMETERS
C  OVER RELAXATION FACTOR, MAXIMUM ITERATIONS, COMPUTER RUN,
C  COMPLETED ITERATION CYCLES, GRID TO USE FLAG
C
      READ 10070, ORF, ITMAX, NRUN, ITP, NGRID
C
C..EDIT MAXIMUM ITERATIONS
C
      IF (ITMAX .GE. 0) GO TO 500
        PRINT 10240
        STOP
```

```
      C
      C..CHECK FOR ANY SPECIFIED SYMMETRY
      C
        500 IF (NSYM .NE. 1) GO TO 600
              JMAX = NBXI
              IMAX = NBET
              MANY = 1
      C
            GO TO 900
        600 IF (NSYM .NE. 2) GO TO 700
              JMAX = NBXI/2
              IMAX = NBET
              MANY = 2
      C
            GO TO 900
        700 IF (NSYM .NE. 3) GO TO 800
              JMAX = NBXI
              IMAX = NBET/2
              MANY = 2
            GO TO 900
      C
        800   JMAX = NBXI/2
              IMAX = NBET/2
              MANY = 4
      C
      C..
      C..MATERIAL PROPERTIES FOR FINE MESH BLOCKS
      C
        900 IBLK = 0
      C
```

SUBROUTINE RDEDIT 74/875 OPT=2 FTN 4.8+670 87/09/01. 13.42.10 PAGE 4

```
      C..INITIALIZE RANDOM ACCESS FILE FOR MATERIAL PROPERTIES
      C
            CALL OPENMS(13,INDX13,KI,0)
            DO  950 NR = 1,NBTFM
        950 CALL WRITMS(13,KODE,25,NR,0)
```

```
C
      DO 2400 NP=1,NSEAM
        NO = (NP-1) * NBLKY * NBLKX * 3
C.. FOR THE NUMBER OF BLOCKS IN THE FINE MESH Y-AXIS
        IV1R = IFYE-IFYS+1
        DO 1300 I=1,IV1R
          IX = I - 1
C.. READ 5 RECORDS(A ROW OF BLOCKS) INTO THE TEMP ARRAY
C
          DO 1000 J=1,5
            READ 10100,(ITMP(J,K),K=1,NELMX)
 1000     CONTINUE
C.. MOVE TEMP ARRAY TO MATERIAL CODE ARRAY
C
         IKOD = 0
         DO 1200 K=1,NELMX
          DO 1100 L=1,5
            LX = 6 - L
            IKOD = IKOD + 1
            KODE(IKOD) = ITMP(LX,K)
 1100     CONTINUE
          IF(IKOD .LT. 25) GO TO 1200
            IKOD = 0
            IMAT = NO + ((IV1R * K/5) - IX) * 3 - 2
            ENCODE(25,10090,MATKOD(IMAT)) KODE
 1200    CONTINUE
 1300  CONTINUE
C.. WRITE OUT MATERIAL PROPERTY CODES TO DISK
C
        M2 = NO + 1
        M3 = (IFXE-IFXS+1) * IV1R * 3 + NO
        DO 2100 M1 = M2,M3,3
           IBLK = IBLK + 1
           DECODE (25,10090,MATKOD(M1)) KODE
C
C   BEGIN CONVERSION TO ARITHMETIC PROPERTY NUMBERS
```

```
C
                DO 2000 J = 1,25
                  I = 1
 1500             IF (KODE(J).EQ.IAMAT(I)) GOTO 1600
                  I = I + 1
                  IF (I.LE.NMATS2) GOTO 1500
C
C   ERROR STOP
C
                  JP = (IBLK / IV1R ) + 1
                  IP = IBLK - (JP - 1) * IV1R
                  PRINT 10250,NP,JP,IP,J,KODE(J),(IAMAT(I),I=1,NMATS2)
                  STOP
C
C
 1600             IF (I-2) 1700,1800,1900
SUBROUTINE RDEDIT 74/875 OPT=2 FTN 4.8+670  87/09/01. 13.42.10  PAGE 5
 1700             MAT(J) = 27
                  GOTO 2000
 1800             MAT(J) = 28
                  GOTO 2000
 1900             MAT(J) = I - 2
 2000           CONTINUE
C
C   STORE PROPERTY NUMBERS FOR THIS BLOCK ON TAPE12
C
                WRITE (12) MAT
 2100   CONTINUE
 2400 CONTINUE
C
C
C    FOR SYMMETRY CASES
C
C    WRITE MATERIAL PROPERTIES FOR ENTIRE MESH
C    TO RANDOM ACCESS FILE 13
C
```

```
      REWIND 12
      IDX = IFXE - IFXS + 1
C
      DO 2500 NP=1,NSEAM
        NO = (NP-1) * NBLKX * NBLKY
C
        DO 2480 JP=1,IDX
          IV2 = (NBLKX + 1) - JP
C
          DO 2460 IP=1,IV1R
            IV1 = (NBLKY+1) - IP
            READ (12) KODE
            IBLK = (JP-1) * NBLKY + IP + NO
            CALL WRITMS(13,KODE,25,IBLK,1)
            IF( NSYM .EQ. 1) GO TO 2460
            IF( NSYM .EQ. 3) GO TO 2420
C
C     NSYM IS 2 OR 4
C
            DO 2410 I=1,5
C
              DO 2410 J=1,5
                IKOD = (I-1) * 5 + J
                IBLK = (5-I) * 5 + J
 2410       MAT(IBLK) = KODE(IKOD)
            IBLK = (IV2-1) * NBLKY + IP + NO
            CALL WRITMS(13,MAT,25,IBLK,1)
            IF( NSYM .NE. 4) GO TO 2460
C
C     NSYM IS 3 OR 4
C
 2420       DO 2430 I=1,5
C
              DO 2430 J=1,5
                IKOD = (I-1) * 5 + J
                IBLK = 5 * I + 1 - J
```

```
 2430       MAT(IBLK) = KODE(IKOD)
            IBLK = (JP-1) * NBLKY + IV1 + NO
SUBROUTINE RDEDIT 74/875 OPT=2 FTN 4.8+670  87/09/01. 13.42.10  PAGE 6
            CALL WRITMS(13,MAT,25,IBLK,1)
            IF(NSYM .NE. 4) GO TO 2460
C
C     NSYM IS 4
C
            DO 2450 I=1,25
              J = 26 - I
 2450       MAT(J) = KODE(I)
            IBLK = (IV2-1) * NBLKY + IV1 + NO
            CALL WRITMS(13,MAT,25,IBLK,1)
 2460     CONTINUE
 2480   CONTINUE
 2500 CONTINUE
C
C
C..MINING STATUS CODES FOR FINE MESH BLOCKS
C
C.. FOR THE NUMBER OF BLOCKS IN THE FINE MESH Y-AXIS
C
      DO 3400 NP=1,NSEAM
        NO = (NP-1) * NBLKY * NBLKX * 3
        DO 3000 I=1,IV1R
           IX = I - 1
C
C.. READ 5 RECORDS(A ROW OF BLOCKS) INTO THE TEMP ARRAY
C
           DO 2600 J=1,5
             READ 10100,(ITMP(J,K),K=1,NELMX)
 2600      CONTINUE
C
C.. MOVE TEMP ARRAY TO MINING CODE ARRAY
C
         IKOD = 0
```

```
               DO 2800 K=1,NELMX
                 DO 2700 L=1,5
                   LX = 6 - L
                   IKOD = IKOD + 1
                   KODE(IKOD) = ITMP(LX,K)
        2700     CONTINUE
                 IF(IKOD .LT. 25) GO TO 2800
                   IKOD = 0
                   IMAT = NO + ((IV1R * K/5) - IX) * 3 - 2
                   ENCODE(25,10090,MKOD(IMAT)) KODE
        2800   CONTINUE
        3000  CONTINUE
        3400 CONTINUE
      C
      C..CHECK TO SEE IF ANY COARSE MESH EXISTS
      C
             IF( NBXI .EQ. IFXE-IFXS+1 .AND. NBET .EQ. IFYE-IFYS+1) GO TO 6000
      C
      C.. READ MATERIAL PROPERTIES FOR COARSE MESH BLOCKS
      C
             DO 4000 NP = 1,NSEAM
               NO = (NP - 1) * NBXE
               DO 3900 IV1 = 1,IMAX
                 IV1R = IMAX + 1 - IV1
SUBROUTINE RDEDIT 74/875 OPT=2 FTN 4.8+670  87/09/01. 13.42.10  PAGE 7
                 READ 10100, (ITMP1(IT), IT=1,JMAX)
                 DO 3800 IV2 = 1,JMAX
                   IDX = (IV2 - 1) * NBET + IV1R + NO
                   MCKOD(IDX) = ITMP1(IV2)
        3800     CONTINUE
        3900   CONTINUE
        4000 CONTINUE
      C
      C.. READ EXTRACTION RATIO CODES FOR COARSE MESH BLOCKS
      C
      C
```

```
      DO 5000 NP = 1,NSEAM
        NO = (NP - 1) * NBXE
        DO 4900 IV1 = 1,IMAX
          IV1R = IMAX + 1 - IV1
          READ 10100, (ITMP1(IT), IT=1,JMAX)
          DO 4800 IV2 = 1,JMAX
            IDX = (IV2 - 1) * NBET + IV1R + NO
            MCEKOD(IDX) = ITMP1(IV2)
 4800     CONTINUE
 4900   CONTINUE
 5000 CONTINUE
C
C..DUPLICATE MATERIAL AND EXTRACTION CODES FOR SYMMETRY
C
      IF(NSYM .EQ. 1) GO TO 6000
C
      DO 5900 NP = 1,NSEAM
        NO = (NP - 1) * NBXE
        DO 5800 IP = 1, IMAX
          IV1 = (NBET + 1) - IP
C COLUMN
          DO 5700 JP = 1, JMAX
            IV2 = (NBXI + 1) - JP
            IV1R = (JP - 1) * NBET + IP + NO
C SYMMETRY CODE 2 OR 4
            IF( NSYM .EQ. 3 ) GO TO 5200
            IDX = (IV2 - 1) * NBET + IP + NO
            MCKOD(IDX) = MCKOD(IV1R)
            MCEKOD(IDX) = MCEKOD(IV1R)
C SYMMETRY CODE 3 OR 4
 5200       IF( NSYM .EQ. 2 ) GO TO 5700
            IDX = (JP - 1) * NBET + IV1 + NO
            MCKOD(IDX) = MCKOD(IV1R)
            MCEKOD(IDX) = MCEKOD(IV1R)
C SYMMETRY CODE 4
            IF( NSYM .NE. 4 ) GO TO 5700
```

```
                 IDX = (IV2 - 1) * NBET + IV1 + NO
                 MCKOD(IDX) = MCKOD(IV1R)
                 MCEKOD(IDX) = MCEKOD(IV1R)
 5700       CONTINUE
 5800     CONTINUE
 5900 CONTINUE
C
C..INPUT FORMAT STATEMENTS
C
10000 FORMAT(20A4)
SUBROUTINE RDEDIT 74/875 OPT=2 FTN 4.8+670  87/09/01. 13.42.10  PAGE 8
10010 FORMAT(F8.2,E12.6,2I8)
10020 FORMAT(2E16.10,4E8.2,I8)
10030 FORMAT(6(F6.0,F6.4))
10040 FORMAT(F8.2,8I8)
10045 FORMAT(I8,9F8.3)
10050 FORMAT(4F8.1)
10060 FORMAT(9F8.5)
10070 FORMAT(F8.2,4I8)
10080 FORMAT(2A10,A5)
10090 FORMAT(25A1)
10100 FORMAT(100A1)
C
C..OUTPUT FORMAT STATEMENTS
C
10200 FORMAT(/* MAXIMUM NUMBER OF SEAMS IS 2*)
10210 FORMAT(/* MAXIMUM NUMBER OF MATERIAL TYPES IS 26*)
10220 FORMAT(/* MAXIMUM NUMBER OF BLOCKS IN EITHER THE X1 OR
     1          THE X2 DIRECTIONS IS 40*)
10230 FORMAT(/* INVALID SYMMETRY CODE(VALID VALUES ARE 1,2,3,4*)
10240 FORMAT(/* MAXIMUM ITERATIONS MUST NOT BE NEGATIVE*)
10250 FORMAT(//* --- ERROR:  BAD MATERIAL PROPERTY CODE ---*/
     #/* SEAM = *,I2/* BLOCK JP = *,I3, * BLOCK IP = *,I3/
     # * ELEMENT *, I3,* PROPERTY CODE IS: *,A1/
     #/* VALID CODES ARE:*/28(1X,A1))
C
```

```
  425           C..RETURN TO MAIN
                C
                 6000 RETURN
                C
                      END
       SYMBOLIC REFERENCE MAP (R=1)
ENTRY POINTS
    3  RDEDIT
VARIABLES          TYPE           RELOCATION
  103  ACC         REAL                CONSTS          0  A11       REAL                IN3
    1  A12         REAL                IN3             2  A13       REAL                IN3
    3  A22         REAL                IN3             4  A23       REAL                IN3
    5  A33         REAL                IN3             0  BW        REAL                IN4
    6  B11         REAL                IN3             7  B12       REAL                IN3
   10  B13         REAL                IN3            11  B22       REAL                IN3
   12  B23         REAL                IN3            13  B33       REAL                IN3
46530  CF          REAL       ARRAY    BK4            70  COHES     REAL       ARRAY    IN2
  160  COHES1      REAL       ARRAY    IN2             6  CON       REAL                CONSTS
    5  COND        REAL                CONSTS          4  CONS      REAL                CONSTS
31000  DE          REAL       ARRAY    BK1             1  DIPANG    REAL                IN6
    0  DUB         REAL       ARRAY    BK1          6200  DVB       REAL       ARRAY    BK1
14400  DWB         REAL       ARRAY    BK1         22600  DWB1      REAL       ARRAY    BK1
   25  E           REAL                IN1             0  EN        REAL       ARRAY    BK3
    0  ES          REAL       ARRAY    IN2            45  EW        REAL                CONSTS
    1  EXR         REAL       ARRAY    IN4A            1  FACT      REAL                CONSTS
       SUBROUTINE RDEDIT 74/875 OPT=2 FTN 4.8+670  87/09/01. 13.42.10  PAGE 9
VARIABLES          TYPE           RELOCATION
   34  GS          REAL       ARRAY    IN2            44  HBW       REAL                CONSTS
   43  HEW         REAL                CONSTS       1416  I         INTEGER
    7  IAMAT       INTEGER    ARRAY    CONSTS       1420  IBLK      INTEGER
   57  ID          INTEGER    ARRAY    CONSTS       1440  IDX       INTEGER
    0  IEX         INTEGER             IN4A            6  IFXE      INTEGER             IN4
    5  IFXS        INTEGER             IN4            10  IFYE      INTEGER             IN4
    7  IFYS        INTEGER             IN4           250  IGOB      INTEGER    ARRAY    IN2
 1427  IKOD        INTEGER                          1432  IMAT      INTEGER
   11  IMAX        INTEGER             BK3         35064  INDEX     INTEGER    ARRAY    BK1
```

36562	INDX13	INTEGER	ARRAY	BK1	1437	IP	INTEGER		
55	IPHASE	INTEGER		CONSTS	1443	IT	INTEGER		
1	ITMAX	INTEGER		IN7	54	ITMAXI	INTEGER		CONSTS
53	ITMINI	INTEGER		CONSTS	1475	ITMP	INTEGER	ARRAY	
2461	ITMP1	INTEGER	ARRAY		3	ITP	INTEGER		IN7
1442	IV1	INTEGER			1424	IV1R	INTEGER		
1441	IV2	INTEGER			1425	IX	INTEGER		
1417	J	INTEGER			12	JMAX	INTEGER		BK3
56	JOG	INTEGER		CONSTS	1436	JP	INTEGER		
1426	K	INTEGER			51	KB	INTEGER		CONSTS
105	KD0	INTEGER		CONSTS	106	KD1	INTEGER		CONSTS
107	KD2	INTEGER		CONSTS	110	KD3	INTEGER		CONSTS
111	KD4	INTEGER		CONSTS	112	KD5	INTEGER		CONSTS
113	KD6	INTEGER		CONSTS	101	KE	INTEGER		CONSTS
100	KI	INTEGER		CONSTS	36531	KODE	INTEGER	ARRAY	BK1
1430	L	INTEGER			77	LQR	INTEGER		CONSTS
1431	LX	INTEGER			0	MANY	INTEGER		F.P.
1444	MAT	INTEGER	ARRAY		0	MATKOD	INTEGER	ARRAY	BK4
12743	MCEKOD	INTEGER	ARRAY	IN9	4543	MCKOD	INTEGER	ARRAY	IN9
0	MKOD	INTEGER	ARRAY	IN9	1435	M1	INTEGER		
1433	M2	INTEGER			1434	M3	INTEGER		
1413	N	INTEGER			2	NBET	INTEGER		IN4
120	NBLKFM	INTEGER		CONSTS	116	NBLKX	INTEGER		CONSTS
117	NBLKY	INTEGER		CONSTS	50	NBT	INTEGER		CONSTS
1414	NBTFM	INTEGER			47	NBXE	INTEGER		CONSTS
1	NBXI	INTEGER		IN4	102	NCUT	INTEGER		CONSTS
114	NELMX	INTEGER		CONSTS	115	NELMY	INTEGER		CONSTS
1415	NFMST	INTEGER			40227	NGOB	INTEGER	ARRAY	BK1
4	NGRID	INTEGER		IN7	27	NMATS	INTEGER		IN1
1412	NMATS2	INTEGER			104	NONE	INTEGER		CONSTS
4	NOSP	INTEGER		IN4	1422	NP	INTEGER		
1421	NR	INTEGER			2	NRUN	INTEGER		IN7
26	NSEAM	INTEGER		IN1	3	NSYM	INTEGER		IN4
52	NTR	INTEGER		CONSTS	1423	NO	INTEGER		
2	OFSANG	REAL		IN6	0	ORF	REAL		IN7
124	PHI	REAL	ARRAY	IN2	214	PHI1	REAL	ARRAY	IN2

```
    0  PI         REAL                 CONSTS          46  RLIM       REAL                 CONSTS
    0  STRANG     REAL                 IN6              0  S13U       REAL        ARRAY    BK4
 6344  S23U       REAL        ARRAY    BK4          23254  S23V       REAL        ARRAY    BK4
14710  S33U       REAL        ARRAY    BK4          31620  S33V       REAL        ARRAY    BK4
40164  S33W       REAL        ARRAY    BK4             14  THIKNS     REAL        ARRAY    IN5
    0  TITLE      REAL        ARRAY    IN1             24  V          REAL                 IN1
    2  V1         REAL                 CONSTS           3  V2         REAL                 CONSTS
    0  XO         REAL        ARRAY    IN5              4  YO         REAL        ARRAY    IN5
   10  ZO         REAL        ARRAY    IN5
       SUBROUTINE RDEDIT 74/875 OPT=2 FTN 4.8+670  87/09/01. 13.42.10  PAGE 10
FILE NAMES        MODE
       INPUT      MIXED                OUTPUT     FMT                     TAPE12     UNFMT
EXTERNALS          TYPE    ARGS
       OPENMS                4                              WRITMS                5
STATEMENT LABELS
   22  100                                 27  200                                   126  300
  133  400                                166  500                                   174  600
  202  700                                211  800                                   215  900
    0  950                                  0  1000                                    0  1100
  315  1200                                 0  1300                                  341  1500
  361  1600                                 0  1700          INACTIVE                365  1800
  367  1900                               371  2000                                    0  2100
    0  2400                                 0  2410                                  460  2420
    0  2430                                 0  2450                                  512  2460
    0  2480                                 0  2500                                    0  2600
    0  2700                               604  2800                                    0  3000
    0  3400                                 0  3800                                    0  3900
    0  4000                                 0  4800                                    0  4900
    0  5000                               745  5200                                  760  5700
    0  5800                                 0  5900                                  770  6000
 1265  10000   FMT                       1267  10010   FMT                          1272  10020   FMT
 1275  10030   FMT                       1300  10040   FMT                          1302  10045   FMT
 1304  10050   FMT                       1306  10060   FMT                          1310  10070   FMT
 1312  10080   FMT  NO REFS              1314  10090   FMT                          1316  10100   FMT
 1320  10200   FMT                       1325  10210   FMT                          1333  10220   FMT
 1346  10230   FMT                       1355  10240   FMT                          1363  10250   FMT
```

```
LOOPS  LABEL   INDEX    FROM-TO    LENGTH   PROPERTIES
   35          N          63 63       17B              EXT REFS
  143          N         125 125      12B              EXT REFS
  221  950     NR        175 176       5B              EXT REFS
  240  2400    NP        178 240     141B              EXT REFS  NOT INNER
  244  1300    I         182 203      60B              EXT REFS  NOT INNER
  247  1000    J         186 188      21B              EXT REFS  NOT INNER
  254          K         187 187      10B              EXT REFS
  273  1200    K         192 202      26B              EXT REFS  NOT INNER
  276  1100    L         193 197       2B    INSTACK
  327  2100    M1        208 239      50B              EXT REFS  NOT INNER
  337  2000    J         214 234      33B              EXT REFS
  410  2500    NP        251 298     112B              EXT REFS  NOT INNER
  414  2480    JP        254 297     103B              EXT REFS  NOT INNER
  421  2460    IP        257 296      74B              EXT REFS  NOT INNER
  441  2410    I         267 272      10B              NOT INNER
  444  2410    J         269 272       2B    INSTACK
  463  2430    I         279 284       6B              NOT INNER
  465  2430    J         281 284       2B    INSTACK
  503  2450    I         291 293       2B    INSTACK
  527  3400    NP        305 331      66B              EXT REFS  NOT INNER
  533  3000    I         307 330      60B              EXT REFS  NOT INNER
  536  2600    J         312 314      21B              EXT REFS  NOT INNER
  543          K         313 313      10B              EXT REFS
  562  2800    K         319 329      26B              EXT REFS  NOT INNER
  565  2700    L         320 324       2B    INSTACK
  633  4000    NP        339 349      23B              EXT REFS  NOT INNER
  636  3900    IV1       341 348      16B              EXT REFS  NOT INNER
       SUBROUTINE RDEDIT 74/875 OPT=2 FTN 4.8+670  87/09/01. 13.42.10  PAGE 11
LOOPS  LABEL   INDEX    FROM-TO    LENGTH   PROPERTIES
  647  3800    IV2       344 347       2B    INSTACK
  664  5000    NP        354 364      23B              EXT REFS  NOT INNER
  667  4900    IV1       356 363      16B              EXT REFS  NOT INNER
  700  4800    IV2       359 362       2B    INSTACK
  723  5900    NP        370 395      45B              NOT INNER
  727  5800    IP        372 394      36B              NOT INNER
```

```
  733  5700     JP          375 393      30B        OPT
COMMON BLOCKS   LENGTH
       BK1       19735
       BK3          11
       BK4       20124
       CONSTS       81
       IN1          24
       IN2         196
       IN3          12
       IN4           9
       IN4A         11
       IN5          16
       IN6           3
       IN7           5
       IN9        8803
STATISTICS
  PROGRAM LENGTH                 2543B     1379
  SCM LABELED COMMON LENGTH  137606B    49030
        60000B SCM USED
```

SUBROUTINE SOLVER 74/875 OPT=2 FTN 4.8+670 87/09/01. 13.42.10 PAGE 1

```
      SUBROUTINE SOLVER (IND1, IND2, IND3, MANY, JTER, IND, ERROR
     #                 ,INDB1,INDB2,INDB3,ERRORB,INDB)
C
C..THIS SUBROUTINE WILL SOLVE FOR THE UNKNOWN CLOSURES AND RIDES
C  THROUGHOUT THE ENTIRE MODEL.
C
      COMMON/BK1/    DUB(3200), DVB(3200), DWB(3200),DWBI(3200),
     #               DE(2100),  INDEX(805),  KODE(25), INDX13(805)
     #               ,NGOB(3200)
      COMMON/BK2/    PO1(4), PO2(4), PO3(4),
     #               P1XI,   P2XI,   P3XI,
     #               P1ET,   P2ET,   P3ET
      COMMON/BK3/    EN(3,3), IMAX, JMAX
      COMMON/BK4/    S13U(3300), S23U(3300), S33U(3300),
     #               S23V(3300), S33V(3300), S33W(3300),
```

```
#               CF(3,27,4)
 COMMON/BK5/    D1U(3300), D2U(3300), D3U(3300),
#               D2V(3300), D3V(3300),
#               D1W(3300), D2W(3300), D3W(3300)
 COMMON/CONSTS/ PI, FACT, V1, V2, CONS, COND, CON, IAMAT(28),
#               HEW, HBW, EW, RLIM, NBXE, NBT, KB, NTR, ITMINI,
#               ITMAXI, IPHASE, JOG, ID(4,4), LQR, KI, KE, NCUT,
#               ACC, NONE, KD0, KD1, KD2, KD3, KD4, KD5, KD6
#               , NELMX, NELMY, NBLKX, NBLKY, NBLKFM
 COMMON/IN1/    TITLE(20), V, E, NSEAM, NMATS
 COMMON/IN2/    ES(28), GS(28), COHES(28),
#               PHI(28), COHES1(28), PHI1(28), IGOB(28)
 COMMON/IN4/    BW, NBXI, NBET, NSYM, NOSP, IFXS, IFXE, IFYS, IFYE
 COMMON/IN4A/   IEX, EXR(10)
 COMMON/IN5/    XO(4), YO(4), ZO(4), THIKNS(4)
 COMMON/IN9/    MKOD(2403), MCKOD(3200), MCEKOD(3200)
 COMMON/AK1/    S13PB, S23PB, S33PB, UPB, VPB, WPB,
#               S13PE, S23PE, S33PE, UPE, VPE, WPE
 COMMON/AK2/    SIG1(25), SIG2(25), SIG3(25),
#               UPOS(25), VPOS(25), WPOS(25),
#               UNEG(25), VNEG(25), WNEG(25)
 LOGICAL COARSE
C
 DIMENSION USBE(12), MAT(25)
 EQUIVALENCE(S13PB,USBE(1))
C
C
 DO 500 ITER=ITMINI,ITMAXI
C
C   REWIND MATERIAL PROPERTY TAPE
C
    REWIND 12
C
    IF (ITER .LT. NCUT) GO TO 50
      KD1=0
      KD2=5
```

```
                KD3=6
                KD4=7
                KD5=8
      C
         50   ERROR=0.0
              ERRORB=0.0
SUBROUTINE SOLVER 74/875 OPT=2 FTN 4.8+670  87/09/01. 13.42.10  PAGE 2

              IND1=0
              IND2=0
              IND3=0
              IBLKNO=0
              ICOUNT=0
              INDB1=0
              INDB2=0
              INDB3=0
      C
              DO 400 NP=1,NSEAM
                THICK=THIKNS(NP)
                LPBO = (NP-1)*NBXE
                LPBOFM = (NP-1) * NBLKFM
                JIP = 1
      C
      C  LOOP FOR EACH BLOCK ON THIS SEAM
      C
                DO 300 JP=1,JMAX
      C
                  DO 200 IP=1,IMAX
      C
      C..SET LOGICAL FOR COARSE MESH BLOCKS
      C
            COARSE = .FALSE.
            IF ( JP .LT. IFXS .OR.
           #   JP .GT. IFXE .OR.
           #   IP .LT. IFYS .OR.
           #   IP .GT. IFYE )  COARSE = .TRUE.
```

```
C
C
C  INITIALIZE THE BLOCK AVERAGE DISPLACEMENT AND STRESS VALUES.
C
            DO 100 JT=1,6
              USBE(JT)=0.0
  100       CONTINUE
C
C  DEFINE THE FINE MESH NEIGHBORHOOD OF THE BLOCK(IP,JP,NP) AT WHICH THE
C  STRESSES ARE BEING COMPUTED.  THE NEIGHBORHOOD CONSISTS OF THE
C  BLOCKS AT THE INTERSECTION OF BLOCK ROWS IB TO IE AND BLOCK COLUMNS
C  JB TO JE
C
            JBF = JP - 1
            JEF = JP + 1
            IBF = IP - 1
            IEF = IP + 1
            IF (JP .LE. IFXS) JBF=IFXS
            IF (JP .GE. IFXE .AND. (NSYM.EQ.1.OR.NSYM.EQ.3)) JEF=IFXE
            IF (IP .LE. IFYS) IBF=IFYS
            IF (IP .GE. IFYE .AND. (NYSM.EQ.1.OR.NSYM.EQ.2)) IEF=IFYE
C
C..ESTABLISH THE TOTAL NEIGHBORHOOD OF THE BLOCK
C
            JB = JP - 1
            JE = JP + 1
            IB = IP - 1
            IE = IP + 1
            IF (JP .EQ. 1) JB = 1
```

SUBROUTINE SOLVER 74/875 OPT=2 FTN 4.8+670 87/09/01. 13.42.10 PAGE 3

```
            IF (JP .EQ. NBXI) JE = NBXI
            IF (IP .EQ. 1) IB = 1
            IF (IP .EQ. NBET) IE = NBET
C
```

```
C  FIRST CONSIDER THE AVERAGE INFLUENCE OF EACH BLOCK NOT IN THE
C  NEIGHBORHOOD OF THIS BLOCK. B-B COEFFICIENTS ARE USED.
C
              CALL BLINFL(IP,JP,NP,INDB1,INDB2,INDB3,THICK,ERRORB,ITER
     #                  ,COARSE,JB,JE,IB,IE)
C
C..SKIP ELEMENT INFLUENCE IF WE ARE PROCESSING A COARSE MESH BLOCK
C
              IF( COARSE ) GO TO 200
C
C
C  READ INTO THE CENTRAL MEMORY THE ELEMENTAL RIDE AND CLOSURE VALUES
C  FOR THE NEIGHBORHOOD OF THE BLOCK (IP,JP,NP)
C
      KIN = 0
      DO 120 NS = 1,NSEAM
        INDIC = ID(NP,NS)
        IF (INDIC .GT. 0) GO TO 120
        LSBO = (NS-1) * NBLKFM
        IF (NS.EQ.NP)
     #        LPR = (KIN+(JP-JBF)*(IEF-IBF+1)+(IP-IBF))*75+1
          DO 110 JR = JBF,JEF
            DO 110 IR = IBF,IEF
              KIN = KIN + 1
              NR = LSBO + (JR - IFXS) * NBLKY + IR - IFYS + 5
              LKINR = (KIN - 1) * 75 + 1
              CALL READMS( 10, DE(LKINR), 75, NR)
  110       CONTINUE
  120 CONTINUE
C
C
C
C   READ ELEMENT PROPERTY NUMBERS
C   FOR THE ELEMENTS IN THIS BLOCK
C
              READ (12) MAT
```

```
      C  UNPACK THE MINING KODES FOR THIS BLOCK.
      C
                   N = (LPBOFM+(JP-IFXS) * (IFYE-IFYS+1)+IP - IFYS) * 3 + 1
                   DECODE (25, 10000, MKOD(N)) KODE
      C
      C
      C  LOOP FOR THE ELEMENTS ON THIS BLOCK, SKIP THE UNMINED ELEMENTS IF THE
      C  SEAM IS CONSIDERED AS INFINITELY RIGID. INITIALIZE THE ELEMENTAL
      C  DISPLACEMENT AND STRESS VALUES BEFORE PROCEEDING WITH COMPUTATIONS.
      C
                   LPB = LPBO + (JP-1) * NBET + IP
                   LPBFM = LPBOFM + (JP - IFXS) * NBLKY + IP - IFYS + 1
                   CALL ELINFL (IP, JP, NP, LPR, MAT, ITER, THICK,
           #                    JB, JE, IB, IE, IND1, IND2, IND3, ERROR
           #                    ,LPE,LPB,LPBFM,IBF,IEF,JBF,JEF)
      C
      C
SUBROUTINE SOLVER 74/875 OPT=2 FTN 4.8+670  87/09/01. 13.42.10  PAGE 4

      C  UPDATE THE MINING KODES FOR THIS BLOCK IN THE PACKED ARRAY, MKODE.
      C  IF THIS IS THE OUTPUT CYCLE, GO TO THE OUTPUT SEGMENT, IF NOT,
      C  UPDATE ON THE MAS STORAGE FILE (TAPE10=DISC), THE ELEMENTAL RIDE
      C  AND CLOSURE VALUES FOR THIS BLOCK (IP,JP,NP).
      C
                   ENCODE (25, 10000, MKOD(N)) KODE
      C
                   IF (IPHASE .EQ. 6HOUTPUT) GO TO 140
                     NRB = LPBFM + 4
                     CALL WRITMS(10, DE(LPR), 75, NRB, 1)
      C
      C  UPDATE ON THE MASS STORAGE FILE (TAPE10=DISC), THE ELEMENTAL RIDE
      C  AND CLOSURE VALUES FOR ALL OTHER SYMMETRICAL BLOCKS.
      C
      C
                     IF( NSYM .NE. 1 )
           #        CALL BLUPDT(IP,JP,NP,LPR,LPBO,LPE,LPB,LPBFM,LPBOFM)
```

```
      GO TO 200
  140 MS=0
      IBLKNO = IBLKNO + 1
      IF(JIP.EQ.1) PRINT  10100,NP
      IF(JIP .EQ. 1) ICOUNT = 0
      IF(JIP .EQ. 1) JIP=2
      UPOSBL = 0.0
      VPOSBL = 0.0
      WPOSBL = 0.0
      UNEGBL = 0.0
      VNEGBL = 0.0
      WNEGBL = 0.0
      SIG1BL = 0.0
      SIG2BL = 0.0
      SIG3BL = 0.0
C
      DO 150 LP=1,5
        DO 150 KP=1,5
          MU = MS + LPR
          MV = MU + 25

          MW = MV + 25
          MS=MS+1
          PRINT 10200,JP,IP,LP,KP,KODE(MS),
     #                DE(MU),DE(MV),DE(MW),
     #                UPOS(MS),VPOS(MS),WPOS(MS),UNEG(MS),
     #                VNEG(MS),WNEG(MS),SIG1(MS),SIG2(MS),
     #                SIG3(MS)
C
         UPOSBL = UPOSBL + UPOS(MS)
         VPOSBL = VPOSBL + VPOS(MS)
         WPOSBL = WPOSBL + WPOS(MS)
         UNEGBL = UNEGBL + UNEG(MS)
         VNEGBL = VNEGBL + VNEG(MS)
         WNEGBL = WNEGBL + WNEG(MS)
         SIG1BL = SIG1BL + SIG1(MS)
```

```
                  SIG2BL = SIG2BL + SIG2(MS)
                  SIG3BL = SIG3BL + SIG3(MS)
C
C
SUBROUTINE SOLVER 74/875 OPT=2 FTN 4.8+670  87/09/01. 13.42.10  PAGE 5

C..WRITE OUT ELEMENT RESULTS TO SEAM FILE
C
                   NUNIT = NP + 30
                   WRITE(NUNIT,10200) JP,IP,LP,KP,KODE(MS),
     #                   DE(MU),DE(MV),DE(MW),
     #                   UPOS(MS),VPOS(MS),WPOS(MS),UNEG(MS),
     #                   VNEG(MS),WNEG(MS),SIG1(MS),SIG2(MS),
     #                   SIG3(MS)
                   ICOUNT = ICOUNT + 1
                   IF (ICOUNT .LT. 55) GO TO 150
                     ICOUNT = 0
                     PRINT 10100,NP
  150            CONTINUE
C
                 UPOSBL = UPOSBL / 25.
                 VPOSBL = VPOSBL / 25.
                 WPOSBL = WPOSBL / 25.
                 UNEGBL = UNEGBL / 25.
                 VNEGBL = VNEGBL / 25.
                 WNEGBL = WNEGBL / 25.
                 SIG1BL = SIG1BL / 25.
                 SIG2BL = SIG2BL / 25.
                 SIG3BL = SIG3BL / 25.
C
                 WRITE(30)  UPOSBL, VPOSBL, WPOSBL,
     #                      UNEGBL, VNEGBL, WNEGBL,
     #                      SIG1BL, SIG2BL, SIG3BL
C
C
  200        CONTINUE
```

```
      C
        300     CONTINUE
      C
        400   CONTINUE
      C
      C  IF ERROR ACCEPTABLE, END THE ITERATION PROCESS.
      C
              JTER=ITER
              IND=IND1+IND2+IND3
              INDB=INDB1+INDB2+INDB3
              IF (IND .LE. 0 .AND. INDB .LE. 0 ) RETURN
      C
        500 CONTINUE
      C
      C
      C..OUTPUT FORMATS
      C
      10000 FORMAT (25I1)
      10100 FORMAT(/*1 NO.OF SEAM--*,I4,//* BLOCK    ELEM  *
           1                           //* COL ROW COL ROW CODE   RIDE1   RIDE
           12 CLOSURE   U1POS   U2POS   U3POS   U1NEG   U2NEG   U3NEG     SIGS
           21     SIGS2      SIGN*/)
      10200 FORMAT(5I4,9F8.3,3F10.2)
            RETURN
            END
   SUBROUTINE SOLVER 74/875 OPT=2 FTN 4.8+670  87/09/01. 13.42.10  PAGE 6
   SYMBOLIC REFERENCE MAP (R=1)
ENTRY POINTS
    3  SOLVER
VARIABLES          TYPE           RELOCATION
  103  ACC        REAL                CONSTS           0  BW        REAL                IN4
46530  CF         REAL        ARRAY   BK4            750  COARSE    LOGICAL
   70  COHES      REAL        ARRAY   IN2            160  COHES1    REAL        ARRAY   IN2
    6  CON        REAL                CONSTS           5  COND      REAL                CONSTS
    4  CONS       REAL                CONSTS       31000  DE        REAL        ARRAY   BK1
    0  DUB        REAL        ARRAY   BK1           6200  DVB       REAL        ARRAY   BK1
```

14400	DWB	REAL	ARRAY	BK1	22600	DWBI	REAL	ARRAY	BK1
0	D1U	REAL	ARRAY	BK5	40164	D1W	REAL	ARRAY	BK5
6344	D2U	REAL	ARRAY	BK5	23254	D2V	REAL	ARRAY	BK5
46530	D2W	REAL	ARRAY	BK5	14710	D3U	REAL	ARRAY	BK5
31620	D3V	REAL	ARRAY	BK5	55074	D3W	REAL	ARRAY	BK5
25	E	REAL		IN1	0	EN	REAL	ARRAY	BK3
0	ERROR	REAL		F.P.	0	ERRORB	REAL		F.P.
0	ES	REAL	ARRAY	IN2	45	EW	REAL		CONSTS
1	EXR	REAL	ARRAY	IN4A	1	FACT	REAL		CONSTS
34	GS	REAL	ARRAY	IN2	44	HBW	REAL		CONSTS
43	HEW	REAL		CONSTS	7	IAMAT	INTEGER	ARRAY	CONSTS
772	IB	INTEGER			766	IBF	INTEGER		
752	IBLKNO	INTEGER			753	ICOUNT	INTEGER		
57	ID	INTEGER	ARRAY	CONSTS	773	IE	INTEGER		
767	IEF	INTEGER			0	IEX	INTEGER		IN4A
6	IFXE	INTEGER		IN4	5	IFXS	INTEGER		IN4
10	IFYE	INTEGER		IN4	7	IFYS	INTEGER		IN4
250	IGOB	INTEGER	ARRAY	IN2	11	IMAX	INTEGER		BK3
0	IND	INTEGER		F.P.	0	INDB	INTEGER		F.P.
0	INDB1	INTEGER		F.P.	0	INDB2	INTEGER		F.P.
0	INDB3	INTEGER		F.P.	35064	INDEX	INTEGER	ARRAY	BK1
776	INDIC	INTEGER			36562	INDX13	INTEGER	ARRAY	BK1
0	IND1	INTEGER		F.P.	0	IND2	INTEGER		F.P.
0	IND3	INTEGER		F.P.	762	IP	INTEGER		
55	IPHASE	INTEGER		CONSTS	1002	IR	INTEGER		
751	ITER	INTEGER			54	ITMAXI	INTEGER		CONSTS
53	ITMINI	INTEGER		CONSTS	770	JB	INTEGER		
764	JBF	INTEGER			771	JE	INTEGER		
765	JEF	INTEGER			760	JIP	INTEGER		
12	JMAX	INTEGER		BK3	56	JOG	INTEGER		CONSTS
761	JP	INTEGER			1001	JR	INTEGER		
763	JT	INTEGER			0	JTER	INTEGER		F.P.
51	KB	INTEGER		CONSTS	105	KD0	INTEGER		CONSTS
106	KD1	INTEGER		CONSTS	107	KD2	INTEGER		CONSTS
110	KD3	INTEGER		CONSTS	111	KD4	INTEGER		CONSTS
112	KD5	INTEGER		CONSTS	113	KD6	INTEGER		CONSTS

```
  101  KE        INTEGER            CONSTS         100  KI        INTEGER             CONSTS
  774  KIN       INTEGER                         36531  KODE      INTEGER    ARRAY    BK1
 1025  KP        INTEGER                          1004  LKINR     INTEGER
 1024  LP        INTEGER                          1006  LPB       INTEGER
 1007  LPBFM     INTEGER                           756  LPBO      INTEGER
  757  LPBOFM    INTEGER                          1010  LPE       INTEGER
 1000  LPR       INTEGER                            77  LQR       INTEGER             CONSTS
  777  LSBO      INTEGER                             0  MANY      INTEGER   *UNUSED    F.P.
       SUBROUTINE SOLVER 74/875 OPT=2 FTN 4.8+670  87/09/01. 13.42.10  PAGE 7
VARIABLES           TYPE          RELOCATION
 1032  MAT       INTEGER  ARRAY                  12743  MCEKOD    INTEGER    ARRAY    IN9
 4543  MCKOD     INTEGER  ARRAY     IN9              0  MKOD      INTEGER    ARRAY    IN9
 1012  MS        INTEGER                          1026  MU        INTEGER
 1027  MV        INTEGER                          1030  MW        INTEGER
 1005  N         INTEGER                             2  NBET      INTEGER             IN4
  120  NBLKFM    INTEGER            CONSTS         116  NBLKX     INTEGER             CONSTS
  117  NBLKY     INTEGER            CONSTS          50  NBT       INTEGER             CONSTS
   47  NBXE      INTEGER            CONSTS           1  NBXI      INTEGER             IN4
  102  NCUT      INTEGER            CONSTS         114  NELMX     INTEGER             CONSTS
  115  NELMY     INTEGER            CONSTS       40227  NGOB      INTEGER    ARRAY    BK1
   27  NMATS     INTEGER            IN1            104  NONE      INTEGER             CONSTS
    4  NOSP      INTEGER            IN4            754  NP        INTEGER
 1003  NR        INTEGER                          1011  NRB       INTEGER
  775  NS        INTEGER                            26  NSEAM     INTEGER             IN1
    3  NSYM      INTEGER            IN4             52  NTR       INTEGER             CONSTS
 1031  NUNIT     INTEGER                           124  PHI       REAL       ARRAY    IN2
  214  PHI1      REAL     ARRAY     IN2              0  PI        REAL                CONSTS
    0  PO1       REAL     ARRAY     BK2              4  PO2       REAL       ARRAY    BK2
   10  PO3       REAL     ARRAY     BK2             17  P1ET      REAL                BK2
   14  P1XI      REAL               BK2             20  P2ET      REAL                BK2
   15  P2XI      REAL               BK2             21  P3ET      REAL                BK2
   16  P3XI      REAL               BK2             46  RLIM      REAL                CONSTS
    0  SIG1      REAL     ARRAY     AK2           1021  SIG1BL    REAL
   31  SIG2      REAL     ARRAY     AK2           1022  SIG2BL    REAL
   62  SIG3      REAL     ARRAY     AK2           1023  SIG3BL    REAL
    0  S13PB     REAL               AK1              6  S13PE     REAL                AK1
```

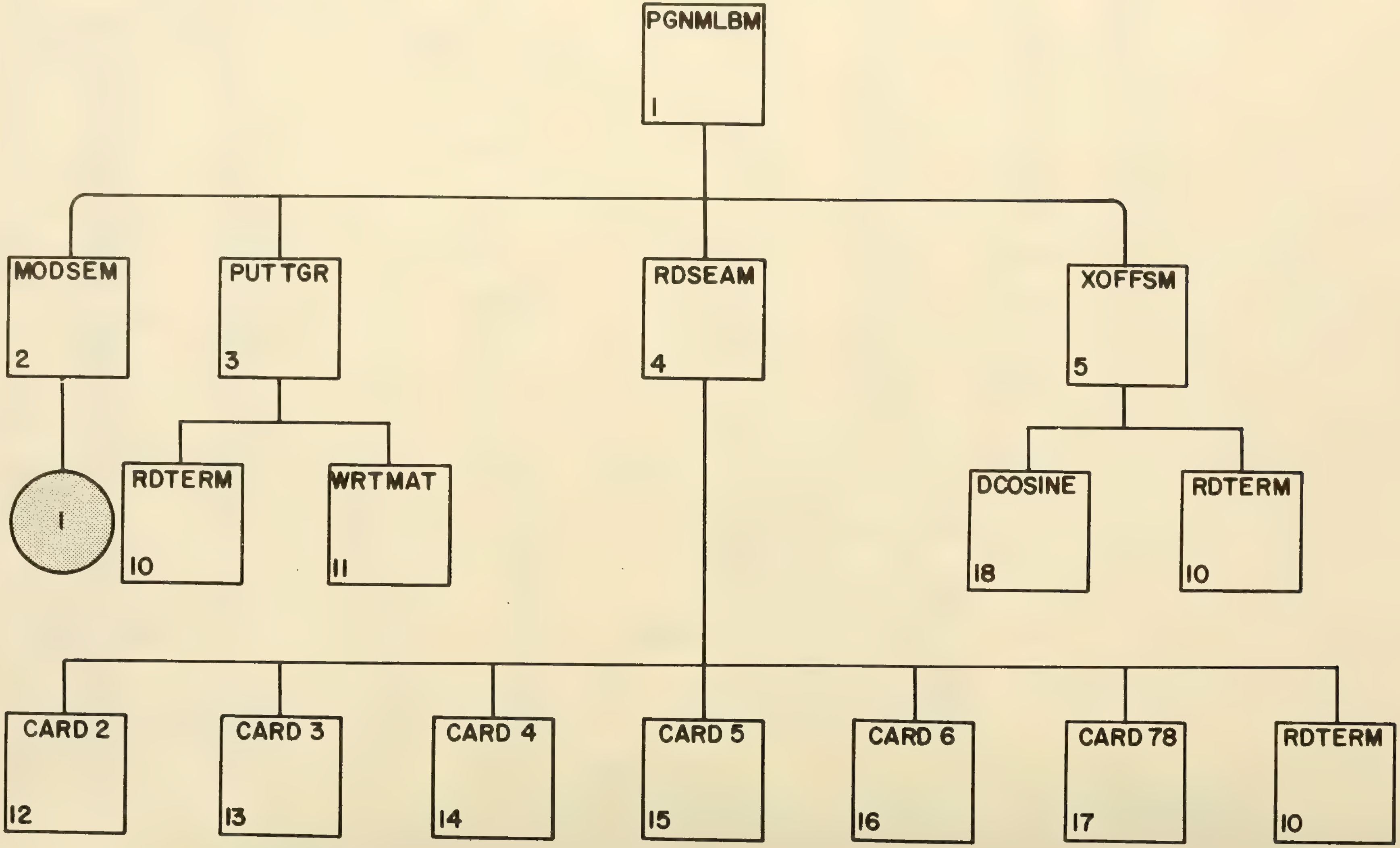
```
      0  S13U       REAL       ARRAY    BK4                1  S23PB     REAL               AK1
      7  S23PE      REAL                AK1             6344  S23U      REAL     ARRAY     BK4
  23254  S23V       REAL       ARRAY    BK4                2  S33PB     REAL               AK1
     10  S33PE      REAL                AK1            14710  S33U      REAL     ARRAY     BK4
  31620  S33V       REAL       ARRAY    BK4            40164  S33W      REAL     ARRAY     BK4
    755  THICK      REAL                                  14  THIKNS    REAL     ARRAY     IN5
      0  TITLE      REAL       ARRAY    IN1              226  UNEG      REAL     ARRAY     AK2
   1016  UNEGBL     REAL                                   3  UPB       REAL               AK1
     11  UPE        REAL                AK1              113  UPOS      REAL     ARRAY     AK2
   1013  UPOSBL     REAL                                   0  USBE      REAL     ARRAY     AK1
     24  V          REAL                IN1              257  VNEG      REAL     ARRAY     AK2
   1017  VNEGBL     REAL                                   4  VPB       REAL               AK1
     12  VPE        REAL                AK1              144  VPOS      REAL     ARRAY     AK2
   1014  VPOSBL     REAL                                   2  V1        REAL               CONSTS
      3  V2         REAL                CONSTS           310  WNEG      REAL     ARRAY     AK2
   1020  WNEGBL     REAL                                   5  WPB       REAL               AK1
     13  WPE        REAL                AK1              175  WPOS      REAL     ARRAY     AK2
   1015  WPOSBL     REAL                                   0  XO        REAL     ARRAY     IN5
      4  YO         REAL       ARRAY    IN5               10  ZO        REAL     ARRAY     IN5
FILE NAMES           MODE
         OUTPUT     FMT                 TAPE12    UNFMT             TAPE30    UNFMT

EXTERNALS            TYPE    ARGS
         BLINFL                14                             BLUPDT             9
         ELINFL                22                             READMS             4
         WRITMS                 5

STATEMENT LABELS
     31  50                                 0  100                                    0  110
    210  120                              266  140                                  430  150
         SUBROUTINE SOLVER 74/875 OPT=2 FTN 4.8+670  87/09/01. 13.42.10  PAGE 8
STATEMENT LABELS
    453  200                                0  300                                    0  400
      0  500                              712  10000   FMT                          714  10100   FMT
    737  10200   FMT
```

```
LOOPS  LABEL    INDEX      FROM-TO    LENGTH    PROPERTIES
   20  500      ITER        43 271     455B               EXT REFS  NOT INNER
   40  400      NP          67 262     423B               EXT REFS  NOT INNER
   46  300      JP          75 260     412B               EXT REFS  NOT INNER
   47  200      IP          77 258     407B               EXT REFS  NOT INNER
   60  100      JT          90 92        2B      INSTACK
  147  120      NS         134 147      45B               EXT REFS  NOT INNER
  164  110      JR         140 146      24B               EXT REFS  NOT INNER
  167  110      IR         141 146      17B               EXT REFS
  312  150      LP         205 241     123B               EXT REFS  NOT INNER
  317  150      KP         206 241     114B               EXT REFS
COMMON BLOCKS   LENGTH
       BK1       19735
       BK2          18
       BK3          11
       BK4       20124
       BK5       26400
       CONSTS       81
       IN1          24
       IN2         196
       IN4           9
       IN4A         11
       IN5          16
       IN9        8803
       AK1          12
       AK2         225
STATISTICS
  PROGRAM LENGTH                 1105B      581
  SCM LABELED COMMON LENGTH    223621B    75665
          60000B SCM USED
```

APPENDIX D.--MULSIM/BM MESH GENERATOR STRUCTURE DIAGRAM AND PROGRAM LISTING



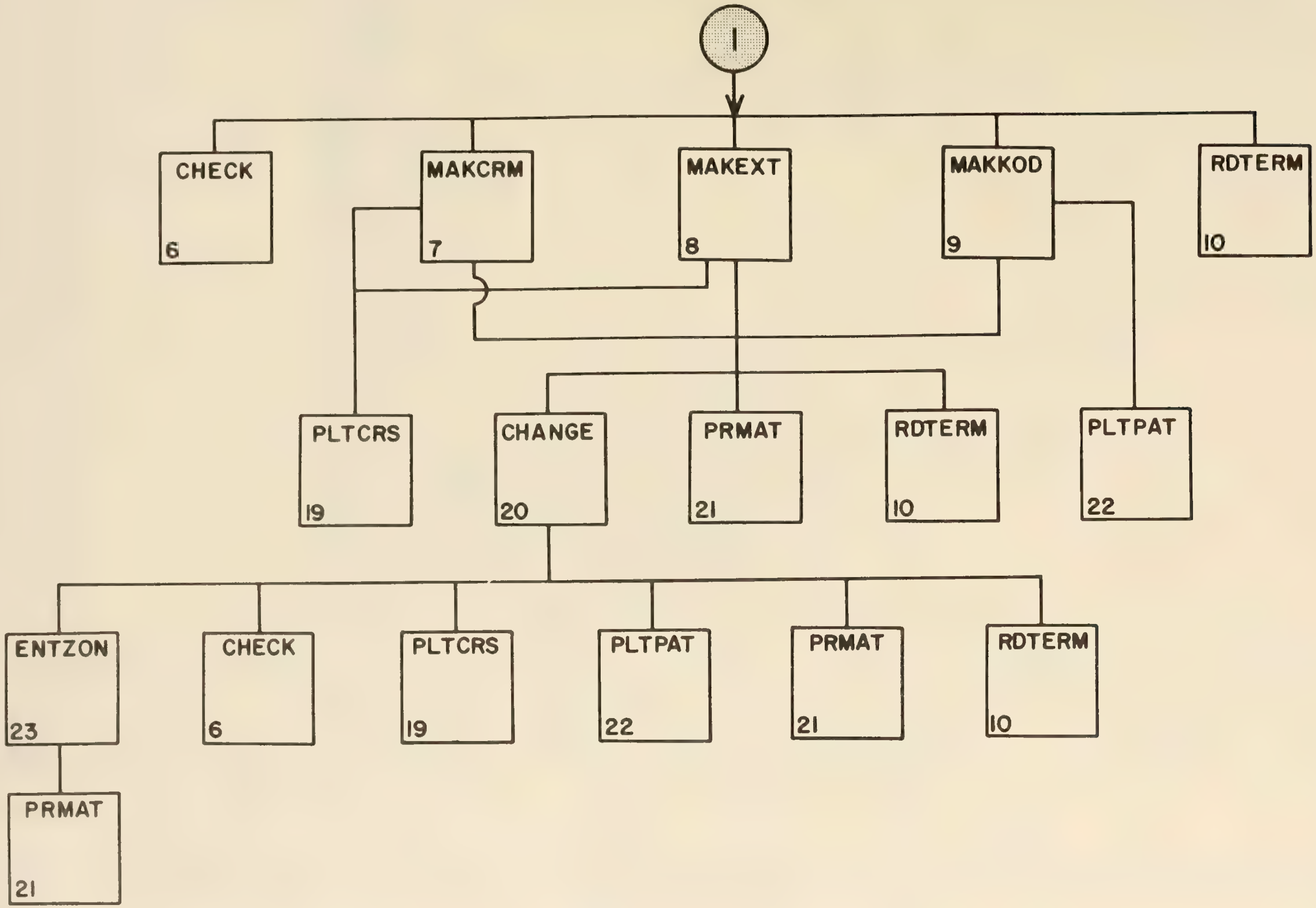


FIGURE D-1.—MULSIM/BM mesh generator structure diagram.

```
   PROGRAM PGNMLBM 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587  86/02/06. 14.17.54  PAGE 1
DO=-LONG/-OT,ARG=-COMMON/-FIXED,CS= USER/-FIXED,DB=-TB/-SB/-SL/-ER/-ID/-PMD/-ST,PL=5000
FTN5,I=PGNMLBM,B=BGNMLBM,L=LGNMLBM,OPT=3.
      C
      C..PROGRAM PGNMLBM
      C..PROGRAM TO GENERATE AN INPUT-DATA FILE FOR MULSIM/BM
      C  THIS VERSION OF PGNMLBM WILL ALLOW THE GENERATION OF
      C  BOUNDARY BLOCKS AND THE MATERIAL PROPERTIES AND EXTRACTION
      C  RATIOS ASSOCIATED WITH THEM
      C
      C
            PROGRAM PGNMLBM(INPUT,OUTPUT,TAPE1,TAPE2,TAPE3,TAPE4)
      C
            CHARACTER USROPT, MAT*10000, SKODE*10000
            CHARACTER  Q, ICONVR, CR*28, KODE*28, GOB*28
            COMMON /CHARS/ Q, ICONVR, CR, KODE, GOB
            COMMON /INTRS/ NSEAM, NX, NY, NOSP, NCNTRL, NMATS, NMATS2,
           #               IFLAG, IFOUND, NXMAX, NYMAX, IWHICH, NXEL, NYEL
           #               ,IFYS,IFYE,IFXS,IFXE,IEX
            DATA CR /'12ABCDEFGHIJKLMNOPQRSTUVWXYZ'/
            DATA NSEAM, NX, NY, NOSP, NCNTRL, NMATS, NMATS2, IFLAG, IFOUND
           #   , NXMAX, NYMAX, IWHICH, NXEL, NYEL /0,0,0,0,0,0,0,0,0,60,60
           #   ,0,0,0/
            OPEN (UNIT=5, FILE='INPUT')
            CALL INITT(120)
            CALL TERM(3,1)
            CALL CHRSIZ(3)
            CALL IOWAIT(15)
            PRINT *,' '
            PRINT *,' '
            PRINT *,'  THIS PROGRAM GENERATES INPUT-DATA FOR MULSIM'
            DO 100 I = 1,999
              PRINT *,' '
              PRINT *,' '
```

```
                  PRINT *,' '
                  PRINT *,'  WHAT DO YOU WANT TO DO NOW'
                  PRINT *,' '
                  PRINT *,'       TYPE 0---TO END THIS PROGRAM NOW'
                  PRINT *,' '
                  PRINT *,'       TYPE 1---TO GENERATE A DATA FILE'
                  CALL RDTERM(1,USROPT,I,R)
                  IF( USROPT .EQ. '0') THEN
                    STOP
                  ELSE IF( USROPT .NE. '1') THEN
                    GO TO 100
                  END IF
                  KODE = CR
                  GOB = CR
                  GOB(1:1) = '0'
                  Q = ' '
                  CALL RDSEAM
                  IF( NOSP .GT. 0 ) CALL XOFFSM
                  NBLOCK = NX * NY
                  NFINE = (IFYE-IFYS+1) * (IFXE-IFXS+1) * 25
                  NSIZE = MAX(NFINE,NBLOCK)
                  CALL MODSEM(MAT(1:NSIZE), SKODE(1:NSIZE), NSIZE)
                  CALL PUTTGR(NBLOCK, MAT(1:NSIZE))
            100 CONTINUE

      PROGRAM PGNMLBM 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587  86/02/06. 14.17.54  PAGE 2
                END

--VARIABLE MAP--(LO=A)
 -NAME---ADDRESS --BLOCK-----PROPERTIES-------TYPE---    -NAME---ADDRESS --BLOCK-----PROPERTIES-------TYPE---
 CR          0B  /CHARS/                      CHAR*28     NFINE    4223B                                INTEGER
 GOB         5B  /CHARS/                      CHAR*28     NMATS       5B  /INTRS/                       INTEGER
 I       4220B                                INTEGER     NMATS2      6B  /INTRS/                       INTEGER
 ICONVR      0B  /CHARS/                      CHAR*1      NOSP        3B  /INTRS/                       INTEGER
 IEX        22B  /INTRS/                      INTEGER     NSEAM       0B  /INTRS/                       INTEGER
 IFLAG       7B  /INTRS/                      INTEGER     NSIZE    4224B                                INTEGER
```

```
 IFOUND      10B  /INTRS/                   INTEGER           NX          1B  /INTRS/           INTEGER
 IFXE        21B  /INTRS/                   INTEGER           NXEL       14B  /INTRS/           INTEGER
 IFXS        20B  /INTRS/                   INTEGER           NXMAX      11B  /INTRS/           INTEGER
 IFYE        17B  /INTRS/                   INTEGER           NY          2B  /INTRS/           INTEGER
 IFYS        16B  /INTRS/                   INTEGER           NYEL       15B  /INTRS/           INTEGER
 IWHICH      13B  /INTRS/                   INTEGER           NYMAX      12B  /INTRS/           INTEGER
 KODE         3B  /CHARS/                   CHAR*28           Q           0B  /CHARS/           CHAR*1
 MAT        300B                            CHAR*10000        R        4221B             *S*    REAL
 NBLOCK    4222B                            INTEGER           SKODE    2250B                    CHAR*10000
 NCNTRL       4B  /INTRS/                   INTEGER           USROPT    277B                    CHAR*1
--PROCEDURES--(LO=A)
 -NAME------TYPE--------ARGS------CLASS-----      -NAME------TYPE--------ARGS------CLASS-----
 CHRSIZ                    1      SUBROUTINE       PUTTGR                   2      SUBROUTINE
 INITT                     1      SUBROUTINE       RDSEAM                   0      SUBROUTINE
 IOWAIT                    1      SUBROUTINE       RDTERM                   4      SUBROUTINE
 MAX        GENERIC      VAR      INTRINSIC        TERM                     2      SUBROUTINE
 MODSEM                    3      SUBROUTINE       XOFFSM                   0      SUBROUTINE
--STATEMENT LABELS--(LO=A)
 -LABEL-ADDRESS-----PROPERTIES----DEF
   100    137B      DO-TERM        55
--ENTRY POINTS--(LO=A)
 -NAME---ADDRESS--ARGS---
 PGNMLBM    23B     0
--I/O UNITS--(LO=A)
 -NAME--- PROPERTIES-------------
 TAPE5   AUX
          PROGRAM PGNMLBM 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587  86/02/06. 14.17.54  PAGE 3
--STATISTICS--

 PROGRAM-UNIT LENGTH            4225B =  2197
 CM LABELLED COMMON LENGTH        34B =  28
 CM STORAGE USED               61100B =  25152
 COMPILE TIME                  0.369 SECONDS
       SUBROUTINE RDSEAM 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587  86/02/06. 14.17.54  PAGE 1
       DO=-LONG/-OT,ARG=-COMMON/-FIXED,CS= USER/-FIXED,DB=-TB/-SB/-SL/-ER/-ID/-PMD/-ST,PL=5000
```

FTN5,I=PGNMLBM,B=BGNMLBM,L=LGNMLBM,OPT=3.

```
C***********************************************************************
C     SUBROUTINE RDSEAM
C***********************************************************************
      SUBROUTINE RDSEAM
      CHARACTER T*80, YN
      CHARACTER  Q, ICONVR, CR*28, KODE*28
      COMMON /CHARS/ Q, ICONVR, CR, KODE
      COMMON /INTRS/ NSEAM, NX, NY, NOSP, NCNTRL, NMATS, NMATS2,
     #               IFLAG, IFOUND, NXMAX, NYMAX, IWHICH, NXEL, NYEL
     #               ,IFYS,IFYE,IFXS,IFXE
C
      OPEN(UNIT=1,FILE='CNTRLS',RECL=80)
C
      CALL NEWPAG
      CALL IOWAIT(15)
      PRINT *,' '
      PRINT *,' '
      PRINT *,' PREPARATION OF INPUT-DATA FOR --ENHANCED MULSIM---'
      PRINT *,'  YOU MUST USE CONSISTENT UNITS FOR:'
      PRINT *,'          DIMENSIONS,    STRESSES,  AND       MODULI '
      PRINT *,' (FOR EXAMPLE: METERS, KILOPASCALS,       KILOPASCALS '
      PRINT *,'              METERS, MEGAPASCALS,       MEGAPASCALS '
      PRINT *,'              INCHES,         PSI,                PSI '
  100 PRINT *,'  DO YOU WANT AUTOMATIC CONVERSION OF DEPTH AND'
      PRINT *,'  DISTANCE ENTERED IN FEET TO INCHES(Y OR N)'
      CALL RDTERM(1,ICONVR,I,R)
      IF (ICONVR .NE. 'Y' .AND. ICONVR .NE. 'N')GO TO 100
      IF (ICONVR .EQ. 'Y') THEN
        PRINT *,'  DEPTH AND DISTANCE VALUES WILL BE CONVERTED'
        PRINT *,'  FROM FEET TO INCHES.'
        PRINT *,'  (MODULUS VALUES WILL NOT BE CONVERTED)'
      ELSE
        PRINT *,'  NO CONVERSION OF UNITS WILL BE DONE.'
      END IF
```

```
      C
      C..READ IN TITLE INFORMATION
      C
        200 PRINT *,' '
            PRINT *,' '
            PRINT *,' CARD TYPE 1--TITLE'
            PRINT *,'  WHAT IS THE TITLE'
            CALL RDTERM(1,T,I,R)
        300 PRINT *,' TITLE ENTERED IS:'
            PRINT *,T
            PRINT *,' IS THIS OK(Y/N)'
            CALL RDTERM(1,YN,I,R)
            IF(YN .NE. 'Y' .AND. YN .NE. 'N') GO TO 300
            IF(YN .EQ. 'N') GO TO 200
            WRITE(1,10000)T
            NCNTRL = 1
      10000 FORMAT(A)
      C
      C..READ IN HOST MATERIAL PROPERTIES
      C
            CALL CARD2

SUBROUTINE RDSEAM 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587  86/02/06. 14.17.54  PAGE 2
      C
      C..READ IN SEAM MATERIAL PROPERTIES
      C
            CALL CARD3
      C
      C..READ IN PRIMITIVE STRESS DATA
      C
            CALL CARD4
      C
      C..READ IN MODEL DATA
      C
            CALL CARD5
      C
```

```
      C..READ IN GLOBAL COORDINATES FOR ALL SEAMS
      C
            CALL CARD6
      C
      C..READ IN GRID ORIENTATION AND RUN FLOW PARAMETERS
      C
            CALL CARD78
            CLOSE (1)
            RETURN
            END
--VARIABLE MAP--(LO=A)
 -NAME---ADDRESS --BLOCK-----PROPERTIES-------TYPE---    -NAME---ADDRESS --BLOCK-----PROPERTIES-------TYPE---

  CR          0B  /CHARS/                     CHAR*28     NMATS2      6B  /INTRS/                     INTEGER
  I         426B                              INTEGER     NOSP        3B  /INTRS/                     INTEGER
  ICONVR      0B  /CHARS/                     CHAR*1      NSEAM       0B  /INTRS/                     INTEGER
  IFLAG       7B  /INTRS/                     INTEGER     NX          1B  /INTRS/                     INTEGER
  IFOUND     10B  /INTRS/                     INTEGER     NXEL       14B  /INTRS/                     INTEGER
  IFXE       21B  /INTRS/                     INTEGER     NXMAX      11B  /INTRS/                     INTEGER
  IFXS       20B  /INTRS/                     INTEGER     NY          2B  /INTRS/                     INTEGER
  IFYE       17B  /INTRS/                     INTEGER     NYEL       15B  /INTRS/                     INTEGER
  IFYS       16B  /INTRS/                     INTEGER     NYMAX      12B  /INTRS/                     INTEGER
  IWHICH     13B  /INTRS/                     INTEGER     Q           0B  /CHARS/                     CHAR*1
  KODE        3B  /CHARS/                     CHAR*28     R         427B                              REAL
  NCNTRL      4B  /INTRS/                     INTEGER     T         415B                              CHAR*80
  NMATS       5B  /INTRS/                     INTEGER     YN        425B                              CHAR*1
--PROCEDURES--(LO=A)
 -NAME------TYPE--------ARGS------CLASS-----     -NAME------TYPE--------ARGS------CLASS-----
  CARD2                    0      SUBROUTINE      CARD78                   0      SUBROUTINE
  CARD3                    0      SUBROUTINE      IOWAIT                   1      SUBROUTINE
  CARD4                    0      SUBROUTINE      NEWPAC                   0      SUBROUTINE
  CARD5                    0      SUBROUTINE      RDTERM                   4      SUBROUTINE
  CARD6                    0      SUBROUTINE

       SUBROUTINE RDSEAM 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587  86/02/06. 14.17.54  PAGE 3
--STATEMENT LABELS--(LO=A)
```

```
-LABEL-ADDRESS-----PROPERTIES----DEF
   100      31B                    24
   200      61B                    38
   300      73B                    43
 10000     253B      FORMAT        51
--ENTRY POINTS--(LO=A)
 -NAME---ADDRESS--ARGS---
  RDSEAM       2B     0
--I/O UNITS--(LO=A)
 -NAME--- PROPERTIES-------------
  TAPE1   AUX/FMT/SEQ
--STATISTICS--
  PROGRAM-UNIT LENGTH             430B =  280
  CM LABELLED COMMON LENGTH        30B =  24
  CM STORAGE USED               61600B =  25472
  COMPILE TIME                  0.409 SECONDS
       SUBROUTINE CARD2 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587  86/02/06. 14.17.54  PAGE 1
       DO=-LONG/-OT,ARG=-COMMON/-FIXED,CS= USER/-FIXED,DB=-TB/-SB/-SL/-ER/-ID/-PMD/-ST,PL=5000
       FTN5,I=PGNMLBM,B=BGNMLBM,L=LGNMLBM,OPT=3.
      C
            SUBROUTINE CARD2
      C
            CHARACTER T*80, YN
            CHARACTER  Q, ICONVR, CR*28, KODE*28
            COMMON /CHARS/ Q, ICONVR, CR, KODE
            COMMON /INTRS/ NSEAM, NX, NY, NOSP, NCNTRL, NMATS, NMATS2,
           #               IFLAG, IFOUND, NXMAX, NYMAX, IWHICH, NXEL, NYEL
           #               ,IFYS,IFYE,IFXS,IFXE
        100 CALL NEWPAG
            CALL IOWAIT(15)
            PRINT *,' '
            PRINT *,' '
            PRINT *,'  CARD TYPE 2---'
        150 PRINT *,'    POISSON''S RATIO OF THE ROCK MASS ='
            CALL RDTERM(3,T,I,V1)
            IF(V1 .LT. 0 .OR. V1 .GT. .5) THEN
```

```
            PRINT *,' POISSONS RATIO MUST BE BETWEEN 0 AND .5'
            GO TO 150
          END IF
      175 PRINT *,' MODULUS OF ELASTICITY OF THE ROCK MASS = '
          CALL RDTERM(3,T,I,E1)
          IF(E1 .LE. 0) THEN
            PRINT *,' MODULUS MUST BE GREATER THAN ZERO'
            GO TO 175
          END IF
          PRINT 10000
    10000 FORMAT( '  THE MAXIMUM GRID AND THE MAXIMUM ',/,
         #        '  NUMBER OF SEAMS ALLOWED DEPEND ',/,
         #        '  ON THE MULSIM/BM VERSION TO BE USED ',/,
         #        '  NUMBER OF SEAMS     GRID MAXIMUM',/,
         #        '       1 - 2            40 X 40  ',/,
         #        '           3            30 X 30  ',/,
         #        '           4            20 X 20  ',/,)
      200 PRINT *,'  NUMBER OF SEAMS ='
          CALL RDTERM(2,T,NSEAM,R)
          IF (NSEAM .LT. 1 .OR. NSEAM .GT. 4) THEN
            PRINT *,' NUMBER OF SEAMS MUST BE 1 THRU 4'
            GO TO 200
          END IF
      300 PRINT *,' HOST MATERIAL PROPERTIES ENTERED ARE:'
          PRINT *,'POISSON"S RATIO: ',V1
          PRINT *,'MODULUS OF ELASTICITY: ',E1
          PRINT *,'NUMBER OF SEAMS: ',NSEAM
          PRINT *,' IS THIS OK(Y/N)'
          CALL RDTERM(1,YN,I,R)
          IF(YN .NE. 'Y' .AND. YN .NE. 'N') GO TO 300
          IF(YN .EQ. 'N') GO TO 100
          NCNTRL = NCNTRL + 1
          WRITE(1,10100)V1, E1, NSEAM
    10100 FORMAT (F8.6,E12.6,I8)
          RETURN
          END
```

```
        SUBROUTINE CARD2 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587  86/02/06. 14.17.54  PAGE 2
--VARIABLE MAP--(LO=A)
 -NAME---ADDRESS --BLOCK-----PROPERTIES-------TYPE---   -NAME---ADDRESS --BLOCK-----PROPERTIES-------TYPE---
  CR            0B  /CHARS/                   CHAR*28    NMATS2       6B  /INTRS/                     INTEGER
  E1          355B                            REAL       NOSP         3B  /INTRS/                     INTEGER
  I           353B                            INTEGER    NSEAM        0B  /INTRS/                     INTEGER
  ICONVR        0B  /CHARS/                   CHAR*1     NX           1B  /INTRS/                     INTEGER
  IFLAG         7B  /INTRS/                   INTEGER    NXEL        14B  /INTRS/                     INTEGER
  IFOUND       10B  /INTRS/                   INTEGER    NXMAX       11B  /INTRS/                     INTEGER
  IFXE         21B  /INTRS/                   INTEGER    NY           2B  /INTRS/                     INTEGER
  IFXS         20B  /INTRS/                   INTEGER    NYEL        15B  /INTRS/                     INTEGER
  IFYE         17B  /INTRS/                   INTEGER    NYMAX       12B  /INTRS/                     INTEGER
  IFYS         16B  /INTRS/                   INTEGER    Q            0B  /CHARS/                     CHAR*1
  IWHICH       13B  /INTRS/                   INTEGER    R          356B                              REAL
  KODE          3B  /CHARS/                   CHAR*28    T          342B                              CHAR*80
  NCNTRL        4B  /INTRS/                   INTEGER    V1         354B                              REAL
  NMATS         5B  /INTRS/                   INTEGER    YN         352B                              CHAR*1
--PROCEDURES--(LO=A)
 -NAME------TYPE--------ARGS------CLASS-----
  IOWAIT                   1      SUBROUTINE
  NEWPAG                   0      SUBROUTINE
  RDTERM                   4      SUBROUTINE
--STATEMENT LABELS--(LO=A)
 -LABEL-ADDRESS-----PROPERTIES----DEF       -LABEL-ADDRESS-----PROPERTIES----DEF
    100      3B                    10          300     52B                     41
    150     15B                    15        10000    163B     FORMAT          28
    175     26B                    21        10100    220B     FORMAT          51
    200     40B                    35
--ENTRY POINTS--(LO=A)
 -NAME---ADDRESS--ARGS---
  CARD2       2B     0
--I/O UNITS--(LO=A)
 -NAME--- PROPERTIES-------------
  TAPE1   FMT/SEQ
--STATISTICS--
  PROGRAM-UNIT LENGTH               357B =  239
```

```
CM LABELLED COMMON LENGTH       30B =  24
CM STORAGE USED              61600B =  25472
COMPILE TIME                 0.385 SECONDS

SUBROUTINE CARD3 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587  86/02/06. 14.17.54  PAGE 1
DO=-LONG/-OT,ARG=-COMMON/-FIXED,CS= USER/-FIXED,DB=-TB/-SB/-SL/-ER/-ID/-PMD/-ST,PL=5000
FTN5,I=PGNMLBM,B=BGNMLBM,L=LGNMLBM,OPT=3.
      C
            SUBROUTINE CARD3
      C
            DIMENSION S1(6)
            CHARACTER T, YN
            CHARACTER  Q, ICONVR, CR*28, KODE*28, GOB*28
            COMMON /CHARS/ Q, ICONVR, CR, KODE, GOB
            COMMON /INTRS/ NSEAM, NX, NY, NOSP, NCNTRL, NMATS, NMATS2,
           #               IFLAG, IFOUND, NXMAX, NYMAX, IWHICH, NXEL, NYEL
           #               ,IFYS,IFYE,IFXS,IFXE
            CALL NEWPAG
            CALL IOWAIT(15)
            PRINT *,' '
            PRINT *,' '
        300 PRINT *,' CARD TYPE 3---'
            PRINT *,'  SEAM MATERIALS ARE SPECIFIED BY A LETTER CODE'
            PRINT *,'   (A,B,C....Z)'
            PRINT *,'  HOW MANY OF THESE MATERIALS ARE THERE'
            CALL RDTERM(2,T,NMATS,R)
            IF (NMATS .LT. 1 .OR. NMATS .GT. 26) GO TO 300
            NCNTRL = NCNTRL + 1
            WRITE (1,10200)NMATS
      10200 FORMAT(I3)
            NMATS2 = NMATS + 2
            DO 405 I = 1, NMATS
        325   T = CR(I+2:I+2)
              CALL NEWPAG
              CALL IOWAIT(15)
              PRINT *,' '
```

```
      PRINT *,' CARD TYPE 4'
      PRINT *,' MATERIAL NUMBER: ---',I
      PRINT *,' MATERIAL CODE:   ---',T
350   PRINT *,' IS THIS A '
      PRINT *,'  GOUGE MATERIAL(MOHR-COULOMB MODEL)?'
      PRINT *,' 3 = "NORMAL MATERIALS" '
      PRINT *,'     (SEAM MATERIALS,GOB, OR "INSERTED" MATERIALS)'
      PRINT *,' 4 = GOUGE MATERIAL (MOHR-COULOMB MODEL)'
      PRINT *,' (ENTER 3 OR 4)'
      CALL RDTERM(1,T,I,R)
      IF (T .NE. '3' .AND. T .NE. '4') GO TO 350
      KODE(I+2:I+2) = T
      PRINT *,'  INPUT MATERIAL PROPERTIES:'
      PRINT *,'   (COHESION AND FRICTION ANGLE CAN BE ASSIGNED 0.0'
      PRINT *,'   IF THEY ARE UNNECESSARY.)'
360   PRINT *,'   YOUNG''S MODULUS ='
      CALL RDTERM(3,T,I,S1(1))
370   PRINT *,'  SHEAR MODULUS ='
      CALL RDTERM(3,T,I,S1(2))
380   PRINT *,'  COHESION OF SOLID MATERIAL ='
      CALL RDTERM(3,T,I,S1(3))
390   PRINT *,'  INTERNAL FRICTION ANGLE OF SOLID MATERIAL ='
      CALL RDTERM(3,T,I,S1(4))
394   PRINT *,'  COHESION OF BROKEN MATERIAL ='
      CALL RDTERM(3,T,I,S1(5))
398   PRINT *,'  INTERNAL FRICTION ANGLE OF BROKEN MATERIAL ='

SUBROUTINE CARD3 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587  86/02/06. 14.17.54  PAGE 2
      CALL RDTERM(3,T,I,S1(6))
400   IGOB = 0
      PRINT *,' IS THIS A GOB/INSERTED MATERIAL TYPE(Y/N)'
      CALL RDTERM(1,YN,I,S1(6))
402   IF(YN.NE. 'Y' .AND.YN .NE. 'N') GO TO 400
      IF(YN .EQ. 'Y') IGOB = 1
      IF(YN .EQ. 'Y') GOB(I+2:I+2) = '0'
404 PRINT *,' DATA ENTERED FOR MATERIAL NUMBER: ',I
```

```
                PRINT *,'  YOUNGS MODULUS: ',S1(1)
                PRINT *,'  SHEAR MODULUS: ',S1(2)
                PRINT *,'  COHESION     : ',S1(3)
                PRINT *,'  INTERNAL ANGLE: ',S1(4)
                PRINT *,'  COHESION BROKEN: ',S1(5)
                PRINT *,'  INTERNAL ANG BROKEN: ',S1(6)
                PRINT *,'  GOB MATERIAL: ',YN
                PRINT *,' IS THIS OK(Y/N)'
                CALL RDTERM(1,YN,I,R)
                IF(YN .NE. 'Y' .AND. YN .NE. 'N') GO TO 404
                IF(YN .EQ. 'N') GO TO 325
                  WRITE(1,10300)S1,IGOB
                  NCNTRL = NCNTRL + 1
          10300 FORMAT(2E16.10,4E8.2,I8)
            405 CONTINUE
                RETURN
                END
--VARIABLE MAP--(LO=A)
 -NAME---ADDRESS --BLOCK-----PROPERTIES-------TYPE---   -NAME---ADDRESS --BLOCK-----PROPERTIES-------TYPE---
  CR          0B  /CHARS/                     CHAR*28    NMATS2       6B  /INTRS/                     INTEGER
  GOB         5B  /CHARS/                     CHAR*28    NOSP         3B  /INTRS/                     INTEGER
  I         755B                              INTEGER    NSEAM        0B  /INTRS/                     INTEGER
  ICONVR      0B  /CHARS/                     CHAR*1     NX           1B  /INTRS/                     INTEGER
  IFLAG       7B  /INTRS/                     INTEGER    NXEL        14B  /INTRS/                     INTEGER
  IFOUND     10B  /INTRS/                     INTEGER    NXMAX       11B  /INTRS/                     INTEGER
  IFXE       21B  /INTRS/                     INTEGER    NY           2B  /INTRS/                     INTEGER
  IFXS       20B  /INTRS/                     INTEGER    NYEL        15B  /INTRS/                     INTEGER
  IFYE       17B  /INTRS/                     INTEGER    NYMAX       12B  /INTRS/                     INTEGER
  IFYS       16B  /INTRS/                     INTEGER    Q            0B  /CHARS/                     CHAR*1
  IGOB      757B                              INTEGER    R          754B                              REAL
  IWHICH     13B  /INTRS/                     INTEGER    S1         744B                              REAL
  KODE        3B  /CHARS/                     CHAR*28    T          752B                              CHAR*1
  NCNTRL      4B  /INTRS/                     INTEGER    YN         753B                              CHAR*1
  NMATS       5B  /INTRS/                     INTEGER
--PROCEDURES--(LO=A)
```

```
-NAME------TYPE--------ARGS------CLASS-----
 IOWAIT                  1       SUBROUTINE
 NEWPAG                  0       SUBROUTINE
 RDTERM                  4       SUBROUTINE

        SUBROUTINE CARD3 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587  86/02/06. 14.17.54  PAGE 3
--STATEMENT LABELS--(LO=A)
 -LABEL-ADDRESS-----PROPERTIES----DEF -LABEL-ADDRESS------DEF   -LABEL-ADDRESS-----PROPERTIES----DEF
   300     13B                     15     380 *NO REFS*     49      402 *NO REFS*                    60
   325     41B                     26     390 *NO REFS*     51      404    203B                      63
   350     63B                     33     394 *NO REFS*     53      405  INACTIVE    DO-TERM         78
   360 *NO REFS*                   45     398 *NO REFS*     55    10200    427B      FORMAT          23
   370 *NO REFS*                   47     400     154B      57    10300    431B      FORMAT          77
--ENTRY POINTS--(LO=A)
 -NAME---ADDRESS--ARGS---
  CARD3       2B     0
--I/O UNITS--(LO=A)
 -NAME--- PROPERTIES-------------
  TAPE1   FMT/SEQ
--STATISTICS--
  PROGRAM-UNIT LENGTH            760B =  496
  CM LABELLED COMMON LENGTH       33B =  27
  CM STORAGE USED              61200B =  25216
  COMPILE TIME                 0.735 SECONDS

        SUBROUTINE CARD4 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587  86/02/06. 14.17.54  PAGE 1
        DO=-LONG/-OT,ARG=-COMMON/-FIXED,CS= USER/-FIXED,DB=-TB/-SB/-SL/-ER/-ID/-PMD/-ST,PL=5000
        FTN5,I=PGNMLBM,B=BGNMLBM,L=LGNMLBM,OPT=3.
            C
                  SUBROUTINE CARD4
            C
                  DIMENSION S2(12)
                  CHARACTER T*80, YN
                  CHARACTER  Q, ICONVR, CR*28, KODE*28
                  COMMON /CHARS/ Q, ICONVR, CR, KODE
                  COMMON /INTRS/ NSEAM, NX, NY, NOSP, NCNTRL, NMATS, NMATS2,
```

```
     #              IFLAG, IFOUND, NXMAX, NYMAX, IWHICH, NXEL, NYEL
     #              ,IFYS,IFYE,IFXS,IFXE
 200 CALL NEWPAG
     CALL IOWAIT(15)
     PRINT *,' '
     PRINT *, ' '
     PRINT *,'  CARD TYPE 5---'
     PRINT *,' INPUT THE PRIMITIVE STRESS DATA'
     PRINT *,'   THE STRESS TENSOR THAT DESCRIBES A STRESS STATE'
     PRINT *,'   THAT EXISTS PRIOR TO MINING IS DEFINED AS LINEAR'
     PRINT *,'   FUNCTIONS OF GLOBAL X3 DIRECTION. WITH COMPRESSION'
     PRINT *,'   BEING POSITIVE, THE EQUATIONS ARE OF THE FORM:'
     PRINT *,'               STRESS(1,1) = A11-B11*X3'
     PRINT *,'               STRESS(1,2) = A12-B12*X3   ETC.'
     PRINT *,'               (X3 IS NEGATIVE)'
     PRINT *,'   AFTER CONVERSION(IF ANY)'
     PRINT *,'     AXX MUST BE >= -.9999 AND <= 99999.'
     PRINT *,'     BXX MUST BE >= -.9999 AND <= 9.9999'
     IF (ICONVR .EQ. 'Y')  THEN
       PRINT *,' INPUT STRESS DATA IN PSI ASSUMING X3 IS IN FEET:'
       PRINT *,' THE MESH GENERATOR WILL DIVIDE YOUR BXX VALUES BY 12'
       PRINT *,' TO MAINTAIN CONSISTENT UNITS SINCE COORDINATES'
       PRINT *,' HAVE BEEN CONVERTED TO INCHES'
     ELSE
       PRINT *,'  INPUT CONSISTENT UNITS; NO CONVERSION WILL BE MADE'
     END IF
 410 PRINT *,'  INPUT A11:'
     CALL RDTERM(3,T,I,S2(1))
     PRINT *, '  INPUT B11: '
     CALL RDTERM(3,T,I,S2(2))
 420 PRINT *,'  INPUT A12:'
     CALL RDTERM(3,T,I,S2(3))
     PRINT *,'  INPUT B12:'
     CALL RDTERM(3,T,I,S2(4))
 430 PRINT *,'  INPUT A13:'
     CALL RDTERM(3,T,I,S2(5))
     PRINT *, '  INPUT B13:'
     CALL RDTERM(3,T,I,S2(6))
```

```
      440 PRINT *,'  INPUT A22:'
          CALL RDTERM(3,T,I,S2(7))
          PRINT *,'  INPUT B22:'
          CALL RDTERM(3,T,I,S2(8))
      450 PRINT *,'  INPUT A23:'
          CALL RDTERM(3,T,I,S2(9))
          PRINT *,'  INPUT B23:'
          CALL RDTERM(3,T,I,S2(10))
      460 PRINT *,'  INPUT A33:'
          CALL RDTERM(3,T,I,S2(11))
          PRINT *,'  INPUT B33:'

SUBROUTINE CARD4 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587  86/02/06. 14.17.54  PAGE 2
          CALL RDTERM(3,T,I,S2(12))
      600 PRINT *,' PRIMITIVE STRESS DATA ENTERED'
          PRINT *,'A11,B11,.......B33'
          PRINT *,S2
          PRINT *,' IS THIS OK(Y/N)'
          CALL RDTERM(1,YN,I,R)
          IF(YN .NE. 'Y' .AND. YN .NE. 'N') GO TO 600
          IF(YN .EQ. 'N') GO TO 200
          IF (ICONVR .EQ. 'Y' ) THEN
            DO 700 I = 2,12,2
              S2(I) = S2(I) / 12
      700   CONTINUE
          END IF
          NCNTRL = NCNTRL + 1
          WRITE (1,10400)S2
    10400 FORMAT(6(F6.0,F6.4))
          RETURN
          END
--VARIABLE MAP--(LO=A)
 -NAME---ADDRESS --BLOCK-----PROPERTIES-------TYPE------NAME---ADDRESS --BLOCK-----PROPERTIES-------TYPE---
  CR          0B  /CHARS/                     CHAR*28   NOSP          3B  /INTRS/                    INTEGER
  I         704B                              INTEGER   NSEAM         0B  /INTRS/                    INTEGER
```

```
 ICONVR       0B  /CHARS/                      CHAR*1   NX           1B  /INTRS/                       INTEGER
 IFLAG        7B  /INTRS/                      INTEGER  NXEL        14B  /INTRS/                       INTEGER
 IFOUND      10B  /INTRS/                      INTEGER  NXMAX       11B  /INTRS/                       INTEGER
 IFXE        21B  /INTRS/                      INTEGER  NY           2B  /INTRS/                       INTEGER
 IFXS        20B  /INTRS/                      INTEGER  NYEL        15B  /INTRS/                       INTEGER
 IFYE        17B  /INTRS/                      INTEGER  NYMAX       12B  /INTRS/                       INTEGER
 IFYS        16B  /INTRS/                      INTEGER  Q            0B  /CHARS/                       CHAR*1
 IWHICH      13B  /INTRS/                      INTEGER  R          705B            *S*                REAL
 KODE         3B  /CHARS/                      CHAR*28  S2         657B                               REAL
 NCNTRL       4B  /INTRS/                      INTEGER  T          673B                               CHAR*80
 NMATS        5B  /INTRS/                      INTEGER  YN         703B                               CHAR*1
 NMATS2       6B  /INTRS/                      INTEGER
--PROCEDURES--(LO=A)
 -NAME------TYPE--------ARGS------CLASS-----
 IOWAIT                    1      SUBROUTINE
 NEWPAG                    0      SUBROUTINE
 RDTERM                    4      SUBROUTINE
--STATEMENT LABELS--(LO=A)
 -LABEL-ADDRESS------DEF        -LABEL-ADDRESS------DEF        -LABEL-ADDRESS-----PROPERTIES----DEF
   200       3B       11           440 *NO REFS*     47           600    141B                     59
   410 *NO REFS*      35           450 *NO REFS*     51           700  INACTIVE    DO-TERM        69
   420 *NO REFS*      39           460 *NO REFS*     55         10400    366B      FORMAT         73
   430 *NO REFS*      43
```

SUBROUTINE CARD4 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587 86/02/06. 14.17.54 PAGE 3

```
--ENTRY POINTS--(LO=A)
 -NAME---ADDRESS--ARGS---
 CARD4       2B     0
--I/O UNITS--(LO=A)
 -NAME--- PROPERTIES-------------
 TAPE1   FMT/SEQ
--STATISTICS--
 PROGRAM-UNIT LENGTH            706B =  454
 CM LABELLED COMMON LENGTH       30B =  24
 CM STORAGE USED              61600B =  25472
 COMPILE TIME                 0.621 SECONDS
```

```
SUBROUTINE CARD5 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587  86/02/06. 14.17.54  PAGE 1
DO=-LONG/-OT,ARG=-COMMON/-FIXED,CS= USER/-FIXED,DB=-TB/-SB/-SL/-ER/-ID/-PMD/-ST,PL=5000
FTN5,I=PGNMLBM,B=BGNMLBM,L=LGNMLBM,OPT=3.
      C
            SUBROUTINE CARD5
      C
            DIMENSION S3(5), EXR(9)
            CHARACTER T*80,MAT*10000, YN
            CHARACTER  Q, ICONVR, CR*28, KODE*28
            COMMON /CHARS/ Q, ICONVR, CR, KODE
            COMMON /INTRS/ NSEAM, NX, NY, NOSP, NCNTRL, NMATS, NMATS2,
           #               IFLAG, IFOUND, NXMAX, NYMAX, IWHICH, NXEL, NYEL
           #               ,IFYS,IFYE,IFXS,IFXE,IEX
        200 CALL NEWPAG
            CALL IOWAIT(15)
            PRINT *,' '
            PRINT *,' '
            PRINT *,'  CARD TYPE 6---'
            PRINT *,'    MODEL DATA'
        510 CALL ANMODE
            PRINT *,' '
            PRINT *,'  IF SYMMETRY IS SPECIFIED, THE AXES OF SYMMETRY'
            PRINT *,'  PASS THROUGH THE CENTER OF THE MODEL'
            PRINT *,' '
            PRINT *,'  WHAT IS THE SYMMETRY CODE'
            PRINT *,'    1 - NO SYMMETRY'
            PRINT *,'    2 - COLUMN SYMMETRY(VERTICAL AXIS OF SYMMETRY)'
            PRINT *,'    3 - ROW SYMMETRY(HORIZONTAL AXIS OF SYMMETRY)'
            PRINT *,'    4 - COLUMN AND ROW SYMMETRY'
            PRINT *,'  ENTER 1,2,3, OR 4'
            CALL RDTERM(3,T,I,S3(4))
            IF( S3(4) .LT. 1 .OR. S3(4) .GT. 4) THEN
              PRINT *,' SYMMETRY CODE MUST BE 1,2,3 OR 4'
              GO TO 510
            END IF
            PRINT *,'    WHAT IS THE BLOCK WIDTH '
```

```
      IF (ICONVR .EQ. 'Y') THEN
         PRINT *,' (INPUT WIDTH IN FEET: YOUR DATA WILL BE CONVERTED'
         PRINT *,' TO INCHES.)'
      END IF
      CALL RDTERM(3,T,I,S3(1))
      IF(S3(1) .LE. 0 .OR. S3(1) .GT. 99999) THEN
        PRINT *,'  BLOCK WIDTH MUST BE > 0 AND < 100000'
        GO TO 510
      END IF
  520 PRINT *,'  HOW MANY BLOCKS LIE ALONG THE LOCAL X AXIS IN '
      PRINT *,'  EACH SEAM (TOTAL GRID: COARSE+FINE)'
      PRINT *,'  IF SYMMETRY WAS SPECIFIED, INPUT ONLY THE UNIQUE'
      PRINT *,'  PORTION (QUADRANT OR HALF) OF THE MESH.'
      CALL RDTERM(3,T,I,S3(2))
      IF( NSEAM .LE. 2 .AND. S3(2) .GT. 40) THEN
        PRINT *,'  NUMBER OF BLOCKS MUST BE LESS THAN 41'
        GO TO 520
      ELSE IF( NSEAM .EQ. 3 .AND. S3(2) .GT. 30) THEN
             PRINT *,'  NUMBER OF BLOCKS MUST BE LESS THAN 31'
             GO TO 520
      ELSE IF( NSEAM .EQ. 4 .AND. S3(2) .GT. 20) THEN
             PRINT *,'  NUMBER OF BLOCKS MUST BE LESS THAN 21'
```



```
             GO TO 520
      ELSE IF(NSEAM .LT. 1) THEN
             PRINT *,' NUMBER OF BLOCKS MUST BE NONZERO'
             GO TO 520
      END IF
      NX = S3(2)
  530 PRINT *,'  HOW MANY BLOCKS LIE ALONG THE LOCAL Y-AXIS'
      PRINT *,'  IN EACH SEAM (TOTAL GRID: COARSE+FINE)'
      PRINT *,'  IF SYMMETRY WAS SPECIFIED, INPUT ONLY THE UNIQUE'
      PRINT *,'  PORTION (QUADRANT OR HALF) OF THE MESH.'
      CALL RDTERM(3,T,I,S3(3))
      IF( NSEAM .LE. 2 .AND. S3(3) .GT. 40) THEN
```

```
          PRINT *,'  NUMBER OF BLOCKS MUST BE LESS THAN 41'
          GO TO 530
        ELSE IF( NSEAM .EQ. 3 .AND. S3(3) .GT. 30) THEN
               PRINT *,'  NUMBER OF BLOCKS MUST BE LESS THAN 31'
               GO TO 530
        ELSE IF( NSEAM .EQ. 4 .AND. S3(3) .GT. 20) THEN
               PRINT *,'  NUMBER OF BLOCKS MUST BE LESS THAN 21'
               GO TO 530
        ELSE IF(NSEAM .LT. 1) THEN
               PRINT *,' NUMBER OF BLOCKS MUST BE NONZERO'
               GO TO 530
        END IF
C       IF(S3(4) .EQ. 2) THEN
C         IF(S3(3) .NE. 2*NX) THEN
C           PRINT *,' MUST BE 2 TIMES NUMBER OF BLOCKS IN X-AXIS'
C           GO TO 520
C         END IF
C       ELSE IF (S3(4) .EQ. 3) THEN
C         IF(S3(3) .NE. NX/2) THEN
C           PRINT *,' MUST BE ONE HALF THE NUMBER OF BLOCKS IN X-AXIS'
C           GO TO 520
C         END IF
C       ELSE IF (S3(4) .EQ. 4) THEN
C         IF(S3(3) .NE. NX) THEN
C            PRINT *,' MUST BE EQUAL TO NUMBER OF BLOCKS IN X-AXIS'
C            GO TO 520
C         END IF
C       END IF
        NY = S3(3)
    600 PRINT *,'  FINE MESH STARTING BLOCK X AXIS'
        CALL RDTERM(2,T,IFXS,R)
        IF( IFXS .LT. 1 .OR. IFXS .GT. NX) THEN
          PRINT *,'  INVALID STARTING BLOCK; REENTER'
          GO TO 600
        END IF
    700 PRINT *,'  FINE MESH ENDING BLOCK X AXIS'
```

```
               CALL RDTERM(2,T,IFXE,R)
               IF( IFXE .LT. 1 .OR. IFXE .GT. NX .OR. IFXE .LT. IFXS) THEN
                 PRINT *,'  INVALID ENDING BLOCK; REENTER'
                 GO TO 700
               END IF
               IF(IFXE-IFXS+1 .GT. 20) THEN
                 PRINT *,' FINE MESH MUST NOT BE GREATER THAN 20 BLOCKS'
                 GO TO 600
               END IF

SUBROUTINE CARD5 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587  86/02/06. 14.17.54  PAGE 3
           800 PRINT *,'  FINE MESH STARTING BLOCK Y AXIS'
               CALL RDTERM(2,T,IFYS,R)
               IF (IFYS .LT. 1 .OR. IFYS .GT. NY) THEN
                 PRINT *,'  INVALID STARTING BLOCK; REENTER'
                 GO TO 800
               END IF
           900 PRINT *,'  FINE MESH ENDING BLOCK Y AXIS'
               CALL RDTERM(2,T,IFYE,R)
               IF( IFYE .LT. 1 .OR. IFYE .GT. NY .OR. IFYE .LT. IFYS) THEN
                 PRINT *,'  INVALID ENDING BLOCK; REENTER'
                 GO TO 900
               END IF
               IF(IFYE-IFYS+1 .GT. 20) THEN
                 PRINT *,' FINE MESH MUST NOT BE GREATER THAN 20 BLOCKS'
                 GO TO 800
               END IF
         C
         C..DISPLAY GRID MODEL AND ALLOW FOR CHANGES
         C
               DO 950 I=1,NX
                 K = (I-1) * NY
                 DO 950 J=1,NY
                   L = K + J
                   MAT(L:L) = ' '
                   IF(I.GE.IFXS .AND. I.LE.IFXE .AND.
```

```
      #      J.GE.IFYS .AND. J.LE.IFYE) MAT(L:L) = '*'
  950 CONTINUE
      CALL PLTCRS(NX,NY,ISEAM,1)
      RXS = IFXS - 1
      RXE = IFXE + 1
      RYS = IFYS - 1
      RYE = IFYE + 1
      CALL PRMAT(MAT,0,RXS,RXE,RYS,RYE,1,0)
  960 CALL HOME
      CALL ANMODE
      PRINT *,' IS MODEL OK(Y/N)'
      CALL RDTERM(1,T,I,R)
      IF(T.NE.'Y'.AND.T.NE.'N') GO TO 960
      IF(T.EQ.'N') THEN
        GO TO 200
      END IF
  965   PRINT *, ' DO YOU WANT'
        PRINT *, ' A HARD COPY(Y/N)'
        CALL RDTERM(1,YN,I,R)
        IF(YN .NE. 'Y' .AND. YN .NE. 'N') GO TO 965
        IF(YN .EQ. 'Y') THEN
          CALL IOWAIT(15)
          CALL HDCOPY
          CALL IOWAIT(25)
        END IF
      CALL NEWPAG
      CALL ANMODE
      PRINT *,' '
      PRINT *,' '
      PRINT *,'  HOW MANY OFF-SEAM PLANES WILL BE USED TO '
      PRINT *,'  OBTAIN ADDITIONAL STRESS AND DISPLACEMENT DATA'
      CALL RDTERM(3,T,I,S3(5))
```

SUBROUTINE CARD5 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587 86/02/06. 14.17.54 PAGE 4

```
  980 PRINT *,' MODEL DATA ENTERED IS:'
      PRINT *,' BLOCK WIDTH: ',S3(1)
```

```
      PRINT *,' X-AXIS BLOCKS: ',S3(2)
      PRINT *,' Y-AXIS BLOCKS: ',S3(3)
      PRINT *,' FINE MESH X  : ',IFXS
      PRINT *,'              : ',IFXE
      PRINT *,' FINE MESH Y  : ',IFYS
      PRINT *,'              : ',IFYE
      PRINT *,' SYMMETRY CODE: ',S3(4)
      PRINT *,' NUM OFFSEAM  : ',S3(5)
      PRINT *,' IS THIS OK(Y/N)'
      CALL RDTERM(1,YN,I,R)
      IF(YN .NE. 'Y' .AND. YN .NE. 'N') GO TO 980
      IF(YN .EQ. 'N') GO TO 200
C     IF (S3(4) .EQ. 4) THEN
C       S3(2) = S3(2) * 2
C       S3(3) = S3(3) * 2
C     ELSE IF (S3(4) .EQ. 3) THEN
C            S3(3) = S3(3) * 2
C         ELSE IF (S3(4) .EQ. 2) THEN
C                S3(2) = S3(2) * 2
C     END IF
      IF (ICONVR .EQ. 'Y') THEN
        S3(1) = S3(1) * 12
      END IF
      NOSP = S3(5)
      IS32 = S3(2)
      IS33 = S3(3)
      IS34 = S3(4)
      IS35 = S3(5)
      WRITE(1,10500)S3(1),IS32,IS33,IS34,IS35,IFXS,IFXE,IFYS,IFYE
10500 FORMAT(F8.2,8I8)
      NCNTRL = NCNTRL + 1
      NXEL = (IFXE - IFXS + 1) * 5
      NYEL = (IFYE - IFYS + 1) * 5
 1000 CALL NEWPAG
      CALL ANMODE
      PRINT *,' '
```

```
                PRINT *,'   CARD TYPE 7---'
                PRINT *,' EXTRACTION CODE "0" IS ZERO EXTRACTION'
                PRINT *,' (THIS CODE IS IRRELEVANT FOR GOB OR OPENINGS)'
                PRINT *,' ENTER NUMBER OF ADDITIONAL'
                PRINT *,' EXTRACTION RATIO CODES(0 TO 9)'
                CALL RDTERM(2,T,IEX,R)
                IF( IEX .LT. 0 .OR. IEX .GT. 9) THEN
                  PRINT *,'  INVALID NUMBER; REENTER'
                  GO TO 1000
                END IF
                IF(IEX.EQ.0) GOTO 1500
                DO 1010 I = 1, IEX
           1005   PRINT *,'  ENTER EXTRACTION RATIO FOR CODE ',I
                  PRINT *,'     (FRACTIONAL NUMBER > 0 AND < 1)'
                  CALL RDTERM(3,T,I,EXR(I))
          C
          C..EDIT RATIO ENTERED
          C   (MUST BE GREATER THAN ZERO AND LESS THAN 1)
          C

SUBROUTINE CARD5 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587  86/02/06. 14.17.54  PAGE 5
                  IF(EXR(I) .LE. 0 .OR. EXR(I) .GE. 1) THEN
                    PRINT *,' RATIO MUST BE GREATER THAN ZERO'
                    PRINT *,' AND LESS THAN 1'
                    GO TO 1005
                  END IF
           1010 CONTINUE
           1500 PRINT *,' EXTRACTION DATA ENTERED IS: '
                PRINT *,'NUMBER OF NON-ZERO EXTRACTION CODES: ',IEX
                IF(IEX .GT. 0) THEN
                  PRINT *,' EXTRACTION RATIOS:',(EXR(IEXR),IEXR=1,IEX)
                ENDIF
                PRINT *,' IS THIS OK(Y/N)'
                CALL RDTERM(1,YN,I,R)
                IF(YN .NE. 'Y' .AND. YN .NE. 'N') GO TO 1500
                IF(YN .EQ. 'N') GO TO 1000
```

```
      10600 FORMAT( I8,9F8.3)
            WRITE(1,10600) IEX, EXR
            NCNTRL = NCNTRL + 1
            RETURN
            END
--VARIABLE MAP--(LO=A)
 -NAME---ADDRESS --BLOCK-----PROPERTIES-------TYPE-----NAME---ADDRESS --BLOCK-----PROPERTIES------TYPE------
  CR          0B  /CHARS/                     CHAR*28  MAT       1744B                             CHAR*10000
  EXR      1723B                              REAL     NCNTRL      4B  /INTRS/                     INTEGER
  I        3715B                              INTEGER  NMATS       5B  /INTRS/                     INTEGER
  ICONVR      0B  /CHARS/                     CHAR*1   NMATS2      6B  /INTRS/                     INTEGER
  IEX        22B  /INTRS/                     INTEGER  NOSP        3B  /INTRS/                     INTEGER
  IEXR     NONE                               INTEGER  NSEAM       0B  /INTRS/                     INTEGER
  IFLAG       7B  /INTRS/                     INTEGER  NX          1B  /INTRS/                     INTEGER
  IFOUND     10B  /INTRS/                     INTEGER  NXEL       14B  /INTRS/                     INTEGER
  IFXE       21B  /INTRS/                     INTEGER  NXMAX      11B  /INTRS/                     INTEGER
  IFXS       20B  /INTRS/                     INTEGER  NY          2B  /INTRS/                     INTEGER
  IFYE       17B  /INTRS/                     INTEGER  NYEL       15B  /INTRS/                     INTEGER
  IFYS       16B  /INTRS/                     INTEGER  NYMAX      12B  /INTRS/                     INTEGER
  ISEAM    3722B              *S*             INTEGER  Q           0B  /CHARS/                     CHAR*1
  IS32     3727B                              INTEGER  R        3716B                              REAL
  IS33     3730B                              INTEGER  RXE      3724B                              REAL
  IS34     3731B                              INTEGER  RXS      3723B                              REAL
  IS35     3732B                              INTEGER  RYE      3726B                              REAL
  IWHICH     13B  /INTRS/                     INTEGER  RYS      3725B                              REAL
  J        3720B                              INTEGER  S3       1716B                              REAL
  K        3717B                              INTEGER  T        1734B                              CHAR*80
  KODE        3B  /CHARS/                     CHAR*28  YN       3714B                              CHAR*1
  L        3721B                              INTEGER
--PROCEDURES--(LO=A)
 -NAME------TYPE--------ARGS------CLASS-----      -NAME------TYPE--------ARGS------CLASS-----
  ANMODE                   0     SUBROUTINE        NEWPAG                   0     SUBROUTINE
  HDCOPY                   0     SUBROUTINE        PLTCRS                   4     SUBROUTINE
  HOME                     0     SUBROUTINE        PRMAT                    8     SUBROUTINE
  IOWAIT                   1     SUBROUTINE        RDTERM                   4     SUBROUTINE
```

```
        SUBROUTINE CARD5 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587  86/02/06. 14.17.54  PAGE 6
--STATEMENT LABELS--(LO=A)
 -LABEL-ADDRESS----DEF        -LABEL-ADDRESS-----PROPERTIES----DEF        -LABEL-ADDRESS-----PROPERTIES----DEF

    200       3B      11          800     246B                      113        1000     534B                   205
    510      17B      17          900     260B                      119        1005     570B                   220
    520      76B      43          950  INACTIVE    DO-TERM          139        1010  INACTIVE    DO-TERM       232
    530     144B      62          960     361B                      146        1500     612B                   233
    600     212B      97          965     403B                      154       10500    1202B     FORMAT        201
    700     224B     103          980     446B                      170       10600    1204B     FORMAT        242
--ENTRY POINTS--(LO=A)
 -NAME---ADDRESS--ARGS---
  CARD5        2B      0
--I/O UNITS--(LO=A)
 -NAME--- PROPERTIES-------------
  TAPE1   FMT/SEQ
--STATISTICS--
  PROGRAM-UNIT LENGTH            3734B =  2012
  CM LABELLED COMMON LENGTH        31B =  25
  CM STORAGE USED               61500B =  25408
  COMPILE TIME                  2.242 SECONDS

        SUBROUTINE CARD6 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587  86/02/06. 14.17.54  PAGE 1
        DO=-LONG/-OT,ARG=-COMMON/-FIXED,CS= USER/-FIXED,DB=-TB/-SB/-SL/-ER/-ID/-PMD/-ST,PL=5000
        FTN5,I=PGNMLBM,B=BGNMLBM,L=LGNMLBM,OPT=3.
      C
            SUBROUTINE CARD6
      C
            DIMENSION S4(4)
            CHARACTER T*80, YN
            CHARACTER  Q, ICONVR, CR*28, KODE*28
            COMMON /CHARS/ Q, ICONVR, CR, KODE
            COMMON /INTRS/ NSEAM, NX, NY, NOSP, NCNTRL, NMATS, NMATS2,
           #               IFLAG, IFOUND, NXMAX, NYMAX, IWHICH, NXEL, NYEL
           #               ,IFYS,IFYE,IFXS,IFXE
            DO 600 I = 1, NSEAM
```

```
500   CALL NEWPAG
      CALL IOWAIT(15)
      PRINT *,' '
      PRINT *,'     CARD TYPE 8---'
      PRINT *,'  SEAM NUMBER: ',I
      PRINT *,'  WHAT ARE THE GLOBAL COORDINATES OF THE GRID'
      PRINT *,'  ORIGIN'
      IF (ICONVR .EQ. 'Y') THEN
        PRINT *,'  (INPUT IN FEET;  YOUR DATA WILL BE CONVERTED'
        PRINT *,'   TO INCHES'
      END IF
552   PRINT *,'    X0 = '
      CALL RDTERM(3,T,I,S4(1))
556   PRINT *,'    Y0 = '
      CALL RDTERM(3,T,I,S4(2))
560   PRINT *,'    Z0 = '
      CALL RDTERM(3,T,I,S4(3))
      IF(S4(3) .GE. 0) THEN
        PRINT *,' MUST BE A NEGATIVE NUMBER'
        GO TO 560
      END IF
564   PRINT *,' THICKNESS = '
      CALL RDTERM(3,T,I,S4(4))
580   PRINT *,' GLOBAL COORDINATES FOR SEAM ',NSEAM
      PRINT *,' X0: ',S4(1)
      PRINT *,' Y0: ',S4(2)
      PRINT *,' Z0: ',S4(3)
      PRINT *,' THICKNESS: ',S4(4)
      PRINT *,' IS THIS OK(Y/N)'
      CALL RDTERM(1,YN,I,R)
      IF(YN .NE. 'Y' .AND. YN .NE. 'N') GO TO 580
      IF(YN .EQ. 'N') GO TO 500
        IF (ICONVR .EQ.'Y') THEN
          DO 590 J = 1,4
590       S4(J) = S4(J) * 12
        END IF
```

```
                  NCNTRL = NCNTRL + 1
                  WRITE (1,10600)S4
          10600 FORMAT(4F8.1)
            600 CONTINUE
                RETURN
                END

        SUBROUTINE CARD6 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587  86/02/06. 14.17.54  PAGE 2
--VARIABLE MAP--(LO=A)
 -NAME---ADDRESS --BLOCK-----------TYPE---------------NAME---ADDRESS --BLOCK-----PROPERTIES-------TYPE---
  CR          0B  /CHARS/         CHAR*28            NMATS2      6B  /INTRS/                      INTEGER
  I         363B                  INTEGER            NOSP        3B  /INTRS/                      INTEGER
  ICONVR      0B  /CHARS/         CHAR*1             NSEAM       0B  /INTRS/                      INTEGER
  IFLAG       7B  /INTRS/         INTEGER            NX          1B  /INTRS/                      INTEGER
  IFOUND     10B  /INTRS/         INTEGER            NXEL       14B  /INTRS/                      INTEGER
  IFXE       21B  /INTRS/         INTEGER            NXMAX      11B  /INTRS/                      INTEGER
  IFXS       20B  /INTRS/         INTEGER            NY          2B  /INTRS/                      INTEGER
  IFYE       17B  /INTRS/         INTEGER            NYEL       15B  /INTRS/                      INTEGER
  IFYS       16B  /INTRS/         INTEGER            NYMAX      12B  /INTRS/                      INTEGER
  IWHICH     13B  /INTRS/         INTEGER            Q           0B  /CHARS/                      CHAR*1
  J         NONE                  INTEGER            R         365B              *S*              REAL
  KODE        3B  /CHARS/         CHAR*28            S4        346B                               REAL
  NCNTRL      4B  /INTRS/         INTEGER            T         352B                               CHAR*80
  NMATS       5B  /INTRS/         INTEGER            YN        362B                               CHAR*1
--PROCEDURES--(LO=A)
 -NAME------TYPE--------ARGS------CLASS-----
  IOWAIT                   1      SUBROUTINE
  NEWPAG                   0      SUBROUTINE
  RDTERM                   4      SUBROUTINE
--STATEMENT LABELS--(LO=A)
 -LABEL-ADDRESS----DEF        -LABEL-ADDRESS----DEF        -LABEL-ADDRESS-----PROPERTIES----DEF
    500      6B     12           560      43B     27           590  INACTIVE    DO-TERM       46
    552 *NO REFS*   23           564 *NO REFS*    33           600  INACTIVE    DO-TERM       51
    556 *NO REFS*   25           580      57B     35         10600      200B    FORMAT        50
--ENTRY POINTS--(LO=A)
```

```
-NAME---ADDRESS--ARGS---
 CARD6       2B     0
--I/O UNITS--(LO=A)
 -NAME--- PROPERTIES-------------
 TAPE1   FMT/SEQ
--STATISTICS--
 PROGRAM-UNIT LENGTH            366B =  246
 CM LABELLED COMMON LENGTH       30B =  24
 CM STORAGE USED              61000B =  25088
 COMPILE TIME                 0.436 SECONDS

SUBROUTINE CARD78 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587  86/02/06. 14.17.54  PAGE 1
DO=-LONG/-OT,ARG=-COMMON/-FIXED,CS= USER/-FIXED,DB=-TB/-SB/-SL/-ER/-ID/-PMD/-ST,PL=5000
FTN5,I=PGNMLBM,B=BGNMLBM,L=LGNMLBM,OPT=3.
C
      SUBROUTINE CARD78
C
      DIMENSION S6(5), DRCOS(3,3)
      CHARACTER T*80, YN
      CHARACTER  Q, ICONVR, CR*28, KODE*28
      COMMON /CHARS/ Q, ICONVR, CR, KODE
      COMMON /INTRS/ NSEAM, NX, NY, NOSP, NCNTRL, NMATS, NMATS2,
     #               IFLAG, IFOUND, NXMAX, NYMAX, IWHICH, NXEL, NYEL
     #               ,IFYS,IFYE,IFXS,IFXE
      CALL NEWPAG
      CALL IOWAIT(15)
      PRINT *,' '
      PRINT *,' '
      PRINT *,'  CARD TYPE 9---'
      PRINT *,'  GRID ORIENTATION'
      CALL DCOSINE(DRCOS)
      NCNTRL = NCNTRL + 1
      WRITE (1,10700)DRCOS
10700 FORMAT(9F8.5)
  610 CALL NEWPAG
      CALL IOWAIT(15)
```

```
      PRINT *,' '
      PRINT *,' '
      PRINT *,'  CARD TYPE 10---'
      PRINT *,'   PROGRAM FLOW PARAMETERS'
      PRINT *,'   THE OVER-RELAXATION FACTOR CAN BE ANY NUMBER'
      PRINT *,'   FROM 1.0 TO 1.99. (1.35 IS OFTEN BEST)'
      PRINT *,'  OVER-RELAXATION FACTOR = '
      CALL RDTERM(3,T,I,S6(1))
      IF( S6(1) .LT. 1 .OR. S6(1) .GT. 1.99) THEN
        PRINT *,' INVALID RELAXATION FACTOR'
        GO TO 610
      END IF
  620 PRINT *,'  WHAT RUN NUMBER IS THIS '
      CALL RDTERM(3,T,I,S6(3))
  630 PRINT *,'  HOW MANY ITERATIONS WERE COMPLETED IN PREVIOUS RUNS'
      CALL RDTERM(3,T,I,S6(4))
  640 PRINT *,'  MAXIMUM NUMBER OF NEW ITERATIONS FOR THIS RUN = '
      CALL RDTERM(3,T,I,S6(2))
      IF(S6(2) .GT. 99999999) S6(2)=99999999
  650 PRINT *,'  IS A PREVIOUS COEFFICIENT MATRIX TO BE USED'
      PRINT *,'  ENTER (1 = YES, 0 = NO) '
      CALL RDTERM(3,T,I,S6(5))
      IF( S6(5) .NE. 0 .AND. S6(5) .NE. 1) GO TO 650
      IS62 = S6(2)
      IS63 = S6(3)
      IS64 = S6(4)
      IS65 = S6(5)
  700 PRINT *,' FLOW PARAMETERS ENTERED ARE:'
      PRINT *,'RELAXATION FACTOR: ',S6(1)
      PRINT *,'RUN NUMBER: ',S6(3)
      PRINT *,'PREV ITERATIONS: ',S6(4)
      PRINT *,'MAX NEW ITERATIONS: ',S6(2)
      PRINT *,'PREV COEFFICIENTS:  ',S6(5)
```

SUBROUTINE CARD78 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587 86/02/06. 14.17.54 PAGE 2

```
      PRINT *,' IS THIS OK(Y/N)'
```

```
                  CALL RDTERM(1,YN,I,R)
                  IF(YN .NE. 'Y' .AND. YN .NE. 'N') GO TO 700
                  IF(YN .EQ. 'N') GO TO 610
                  WRITE (1,10800)S6(1),IS62,IS63,IS64,IS65
            10800 FORMAT(F8.2,4I8)
                  NCNTRL = NCNTRL + 1
                  RETURN
                  END
--VARIABLE MAP--(LO=A)
 -NAME---ADDRESS --BLOCK-------TYPE-----                 -NAME---ADDRESS --BLOCK-----PROPERTIES-------TYPE---
  CR          0B  /CHARS/     CHAR*28                      NCNTRL       4B  /INTRS/                    INTEGER
  DRCOS     464B              REAL                         NMATS        5B  /INTRS/                    INTEGER
  I         506B              INTEGER                      NMATS2       6B  /INTRS/                    INTEGER
  ICONVR      0B  /CHARS/     CHAR*1                       NOSP         3B  /INTRS/                    INTEGER
  IFLAG       7B  /INTRS/     INTEGER                      NSEAM        0B  /INTRS/                    INTEGER
  IFOUND     10B  /INTRS/     INTEGER                      NX           1B  /INTRS/                    INTEGER
  IFXE       21B  /INTRS/     INTEGER                      NXEL        14B  /INTRS/                    INTEGER
  IFXS       20B  /INTRS/     INTEGER                      NXMAX       11B  /INTRS/                    INTEGER
  IFYE       17B  /INTRS/     INTEGER                      NY           2B  /INTRS/                    INTEGER
  IFYS       16B  /INTRS/     INTEGER                      NYEL        15B  /INTRS/                    INTEGER
  IS62      507B              INTEGER                      NYMAX       12B  /INTRS/                    INTEGER
  IS63      510B              INTEGER                      Q            0B  /CHARS/                    CHAR*1
  IS64      511B              INTEGER                      R          513B            *S*              REAL
  IS65      512B              INTEGER                      S6         457B                             REAL
  IWHICH     13B  /INTRS/     INTEGER                      T          475B                             CHAR*80
  KODE        3B  /CHARS/     CHAR*28                      YN         505B                             CHAR*1
--PROCEDURES--(LO=A)
 -NAME------TYPE--------ARGS------CLASS-----
  DCOSINE                  1      SUBROUTINE
  IOWAIT                   1      SUBROUTINE
  NEWPAG                   0      SUBROUTINE
  RDTERM                   4      SUBROUTINE
--STATEMENT LABELS--(LO=A)
 -LABEL-ADDRESS-----PROPERTIES----DEF       -LABEL-ADDRESS-----PROPERTIES----DEF
    610     24B                     21          650     75B                     42
    620 *NO REFS*                   35          700    115B                     50
```

```
  630 *NO REFS*                37        10700     263B      FORMAT          20
  640 *NO REFS*                39        10800     265B      FORMAT          61

      SUBROUTINE CARD78 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587  86/02/06. 14.17.54  PAGE 3
--ENTRY POINTS--(LO=A)
 -NAME---ADDRESS--ARGS---
  CARD78      2B     0
--I/O UNITS--(LO=A)
 -NAME--- PROPERTIES-------------
  TAPE1   FMT/SEQ
--STATISTICS--
  PROGRAM-UNIT LENGTH            514B =  332
  CM LABELLED COMMON LENGTH       30B =  24
  CM STORAGE USED              61100B =  25152
  COMPILE TIME                 0.492 SECONDS

      SUBROUTINE DCOSINE 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587  86/02/06. 14.17.54  PAGE 1
      DO=-LONG/-OT,ARG=-COMMON/-FIXED,CS= USER/-FIXED,DB=-TB/-SB/-SL/-ER/-ID/-PMD/-ST,PL=5000
      FTN5,I=PGNMLBM,B=BGNMLBM,L=LGNMLBM,OPT=3.
C**********************************************************************
C     SUBROUTINE DCOSINE
C**********************************************************************
      SUBROUTINE DCOSINE(DRCOS)
      DIMENSION DRCOS(3,3)
      CHARACTER T, YN
C
    5 PRINT *,' '
      PRINT *,' '
      PRINT *,' SPECIFY STRIKE, DIP AND OFFSET ANGLE'
      PRINT *,' STRIKE ANGLE IS IN DEGREES COUNTERCLOCKWISE FROM'
      PRINT *,' GLOBAL Y'
      PRINT *,' (COUNTERCLOCKWISE IS MEASURED LOOKING DOWN FROM THE'
      PRINT *,' SURFACE)'
   10 PRINT *,' WHAT IS THE STRIKE ANGLE'
      CALL RDTERM(3,T,I,S)
      IF( S .LT. -360 .OR. S .GT. 360) THEN
```

```
      PRINT *,'INVALID ANGLE'
      GO TO 10
    END IF
 20 PRINT *,', POSITIVE DIP IS TOWARD THE ROTATED X DIRECTION.'
    PRINT *,'WHAT IS THE ANGLE OF DIP?'
    CALL RDTERM(3,T,I,D)
    IF( D .LT. -90 .OR. D .GT. 90) THEN
      PRINT *,'INVALID ANGLE'
      GO TO 20
    END IF
 30 PRINT *,' IN THE PLANE OF THE SEAM, THE OFFSET ANGLE'
    PRINT *,' IS MEASURED COUNTERCLOCKWISE FROM THE DIP DIRECTION.'
    PRINT *,' TO THE ENTRY LINE (THE LOCAL X-AXIS OF YOUR MESH)'
    PRINT *,' WHAT IS YOUR OFFSET ANGLE?'
    CALL RDTERM(3,T,I,R)
    IF( R .LT. -360 .OR. R .GT. 360) THEN
      PRINT *,'INVALID ANGLE'
      GO TO 30
    END IF
C
 40 PRINT *,' ANGLES ENTERED ARE: '
    PRINT *,'  STRIKE: ',S
    PRINT *,'     DIP: ',D
    PRINT *,'  OFFSET: ',R
    PRINT *,' IS THIS OK(Y/N)'
    CALL RDTERM(1,YN,I,R)
    IF(YN .NE. 'Y' .AND. YN .NE. 'N') GO TO 40
    IF(YN .EQ. 'N') GO TO 5
C..CALCULATE DIRECTION COSINES FROM STRIKE,DIP AND OFFSET ANGLES
C
    DRCOS(1,1) = COSD(S) * COSD(D) * COSD(R) -
   #          SIND(S) * SIND(R)
    DRCOS(2,1) = -COSD(S) * COSD(D) * SIND(R) -
   #          SIND(S) * COSD(R)
    DRCOS(3,1) = COSD(S) * SIND(D)
    DRCOS(1,2) = SIND(S) * COSD(D) * COSD(R) +
```

```
                #          COSD(S) * SIND(R)
                 DRCOS(2,2) = -SIND(S) * COSD(D) * SIND(R) +

   SUBROUTINE DCOSINE 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587  86/02/06. 14.17.54  PAGE 2
                #          COSD(S) * COSD(R)
                 DRCOS(3,2) = SIND(S) * SIND(D)
                 DRCOS(1,3) = -SIND(D) * COSD(R)
                 DRCOS(2,3) = SIND(D) * SIND(R)
                 DRCOS(3,3) = COSD(D)
           C
                 RETURN
                 END
--VARIABLE MAP--(LO=A)
 -NAME---ADDRESS --PROPERTIES-----------------TYPE---            -NAME---ADDRESS -- -TYPE--
  D          476B                             REAL                S           475B    REAL
  DRCOS        1  DUMMY-ARG                   REAL                T           472B    CHAR*1
  I          474B                             INTEGER             YN          473B    CHAR*1
  R          477B                             REAL
--PROCEDURES--(LO=A)
 -NAME------TYPE--------ARGS------CLASS-----
  COSD      REAL           1      INTRINSIC
  RDTERM                   4      SUBROUTINE
  SIND      REAL           1      INTRINSIC
--STATEMENT LABELS--(LO=A)
 -LABEL-ADDRESS-----PROPERTIES----DEF
      5     5B                       8
     10    23B                      15
     20    35B                      21
     30    51B                      28
     40    71B                      38
--ENTRY POINTS--(LO=A)
 -NAME---ADDRESS--ARGS---
  DCOSINE     3B     1
--STATISTICS--
  PROGRAM-UNIT LENGTH            500B =  320
  CM STORAGE USED              61000B =  25088
```

COMPILE TIME 0.591 SECONDS

```
SUBROUTINE XOFFSM 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587  86/02/06. 14.17.54  PAGE 1
DO=-LONG/-OT,ARG=-COMMON/-FIXED,CS= USER/-FIXED,DB=-TB/-SB/-SL/-ER/-ID/-PMD/-ST,PL=5000
FTN5,I=PGNMLBM,B=BGNMLBM,L=LGNMLBM,OPT=3.

C*********************************************************************
C     SUBROUTINE XOFFSEAM
C*********************************************************************
      SUBROUTINE XOFFSM
      CHARACTER  Q, ICONVR, CR*28, KODE*28, YN
      COMMON /CHARS/ Q, ICONVR, CR, KODE
      COMMON /INTRS/ NSEAM, NX, NY, NOSP, NCNTRL, NMATS, NMATS2,
     #               IFLAG, IFOUND, NXMAX, NYMAX, IWHICH, NXEL, NYEL
     #               ,IFYS,IFYE,IFXS,IFXE
      DIMENSION DRCOS(3,3), S7(7)
      OPEN (UNIT = 3, FILE = 'OFSM', RECL = 80)
      DO 100 I = 1, NOSP
   10   CALL NEWPAG
        CALL IOWAIT(15)
        PRINT *,' '
        PRINT *,' '
        PRINT *, '  CARD TYPE 15-'
        PRINT *, '  OFFSEAM GRID DETAILS: PLANE NO. ',I
        IF (ICONVR .EQ. 'Y') THEN
          PRINT *,' INPUT COORDINATES AND WIDTH IN FEET;'
          PRINT *,' AND THEY WILL BE CONVERTED TO INCHES'
        END IF
        PRINT *,'     LOCAL ORIGIN'
        PRINT *,'            XO='
        CALL RDTERM(3,T,I,S7(1))
   20   PRINT *,'            YO='
        CALL RDTERM(3,T,I,S7(2))
   30   PRINT *,'            ZO='
        CALL RDTERM(3,T,I,S7(3))
   40   PRINT *,' BLOCK WIDTH ALONG THE LOCAL X AXIS='
```

```
      CALL RDTERM(3,T,I,S7(4))
45    PRINT *,' BLOCK WIDTH ALONG THE LOCAL Y AXIS='
      CALL RDTERM(3,T,I,S7(5))
55    PRINT *,' NUMBER OF BLOCKS IN LOCAL X DIRECTION'
      CALL RDTERM(3,T,I,S7(6))
      IF( S7(6) .LT. 1 .OR. S7(6) .GT. 20) THEN
        PRINT *,'NUMBER OF BLOCKS MUST BE 1 THRU 20'
        GO TO 55
      END IF
60    PRINT *,' NUMBER OF BLOCKS IN LOCAL Y DIRECTION'
      CALL RDTERM(3,T,I,S7(7))
      IF( S7(7) .LT. 1 .OR. S7(6) .GT. 20) THEN
        PRINT *,'NUMBER OF BLOCKS MUST BE 1 THRU 20'
        GO TO 60
      END IF
64    IS76 = S7(6)
      IS77 = S7(7)
80    PRINT *,' OFFSEAM GRID DETAILS FOR PLANE NO: ',I
      PRINT *,' XO: ',S7(1)
      PRINT *,' YO: ',S7(2)
      PRINT *,' ZO: ',S7(3)
      PRINT *,' X-AXIS BLOCK WIDTH: ',S7(4)
      PRINT *,' Y-AXIS BLOCK WIDTH: ',S7(5)
      PRINT *,' X-AXIS NUM OF BLOCKS: ',S7(6)
      PRINT *,' Y-AXIS NUM OF BLOCKS: ',S7(7)

SUBROUTINE XOFFSM 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587  86/02/06. 14.17.54  PAGE 2
      PRINT *,' IS THIS OK(Y/N)'
      CALL RDTERM(1,YN,I,R)
      IF(YN .NE. 'Y' .AND. YN .NE. 'N') GO TO 80
      IF(YN .EQ. 'N') GO TO 10
      IF (ICONVR .EQ. 'Y') THEN
        DO 90 J = 1,5
          S7(J) = S7(J) * 12
90      CONTINUE
      END IF
```

```
                WRITE(3,10000) (S7(IS7),IS7=1,5),IS76,IS77
      10000 FORMAT(5E12.6,2I8)
                PRINT *, ' CARD TYPE 15A--'
                PRINT *,'  INPUT PLANE ORIENTATION'
                CALL DCOSINE(DRCOS)
                WRITE(3,10100) DRCOS
      10100 FORMAT(9F8.5)
        100 CONTINUE
            CLOSE (3)
            RETURN
            END
--VARIABLE MAP--(LO=A)
 -NAME---ADDRESS --BLOCK-------TYPE---------       -NAME---ADDRESS --BLOCK-----PROPERTIES-------TYPE---
  CR          0B  /CHARS/      CHAR*28              NCNTRL       4B  /INTRS/                     INTEGER
  DRCOS     556B               REAL                 NMATS        5B  /INTRS/                     INTEGER
  I         576B               INTEGER              NMATS2       6B  /INTRS/                     INTEGER
  ICONVR      0B  /CHARS/      CHAR*1               NOSP         3B  /INTRS/                     INTEGER
  IFLAG       7B  /INTRS/      INTEGER              NSEAM        0B  /INTRS/                     INTEGER
  IFOUND     10B  /INTRS/      INTEGER              NX           1B  /INTRS/                     INTEGER
  IFXE       21B  /INTRS/      INTEGER              NXEL        14B  /INTRS/                     INTEGER
  IFXS       20B  /INTRS/      INTEGER              NXMAX       11B  /INTRS/                     INTEGER
  IFYE       17B  /INTRS/      INTEGER              NY           2B  /INTRS/                     INTEGER
  IFYS       16B  /INTRS/      INTEGER              NYEL        15B  /INTRS/                     INTEGER
  IS7       604B               INTEGER              NYMAX       12B  /INTRS/                     INTEGER
  IS76      601B               INTEGER              Q            0B  /CHARS/                     CHAR*1
  IS77      602B               INTEGER              R          603B              *S*             REAL
  IWHICH     13B  /INTRS/      INTEGER              S7         567B                              REAL
  J         NONE               INTEGER              T          600B                              REAL
  KODE        3B  /CHARS/      CHAR*28              YN         555B                              CHAR*1
--PROCEDURES--(LO=A)
 -NAME------TYPE--------ARGS------CLASS-----
  DCOSINE                 1       SUBROUTINE
  IOWAIT                  1       SUBROUTINE
  NEWPAG                  0       SUBROUTINE
  RDTERM                  4       SUBROUTINE
```

```
        SUBROUTINE XOFFSM 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587  86/02/06. 14.17.54  PAGE 3
--STATEMENT LABELS--(LO=A)
 -LABEL-ADDRESS----DEF        -LABEL-ADDRESS----DEF        -LABEL-ADDRESS-----PROPERTIES----DEF
    10      11B      13            55      62B      34            90  INACTIVE   DO-TERM         63
    20 *NO REFS*    26            60      74B      40           100  INACTIVE   DO-TERM         72
    30 *NO REFS*    28            64 *NO REFS*    46         10000     317B     FORMAT          66
    40 *NO REFS*    30            80     112B      48         10100     322B     FORMAT          71
    45 *NO REFS*    32
--ENTRY POINTS--(LO=A)
 -NAME---ADDRESS--ARGS---
  XOFFSM      2B     0
--I/O UNITS--(LO=A)
 -NAME--- PROPERTIES-------------
  TAPE3   AUX/FMT/SEQ
--STATISTICS--
  PROGRAM-UNIT LENGTH              605B =  389
  CM LABELLED COMMON LENGTH         30B =  24
  CM STORAGE USED                61200B =  25216
  COMPILE TIME                   0.591 SECONDS

        SUBROUTINE MODSEM 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587  86/02/06. 14.17.54  PAGE 1
        DO=-LONG/-OT,ARG=-COMMON/-FIXED,CS= USER/-FIXED,DB=-TB/-SB/-SL/-ER/-ID/-PMD/-ST,PL=5000
        FTN5,I=PGNMLBM,B=BGNMLBM,L=LGNMLBM,OPT=3.
                  C*********************************************************************
                  C     SUBROUTINE MODSEM
                  C*********************************************************************
                        SUBROUTINE MODSEM(MAT, SKODE, NSIZE)
                        CHARACTER *(*) MAT, SKODE
                        CHARACTER FC, VR, TEMP*6,P, A, S*2, GOB*28
                        CHARACTER  Q, ICONVR, CR*28, KODE*28
                        COMMON /CHARS/ Q, ICONVR, CR, KODE, GOB
                        COMMON /INTRS/ NSEAM, NX, NY, NOSP, NCNTRL, NMATS, NMATS2,
                       #               IFLAG, IFOUND, NXMAX, NYMAX, IWHICH, NXEL, NYEL
                       #               ,IFYS,IFYE,IFXS,IFXE
                        FC = ' '
                        VR = ' '
```

```
      DO 400 ISEAM = 1, NSEAM
  100   CALL NEWPAG
        CALL IOWAIT(15)
        CALL ANMODE
  110   PRINT 115, ISEAM
  115 FORMAT(//'  THE FINE MESH AREA WILL BE MODELED FIRST'/
     #         '  FOLLOWED BY THE COARSE MESH AREA'//
     #         '  ENTER THE DEFAULT PROPERTY FOR SEAM ',I5)
        CALL RDTERM(1,P,I,R)
        CALL CHECK(P)
        IF(IFLAG .EQ. 0) GO TO 100
        Q = KODE(IWHICH:IWHICH)
        IGOBIN = IWHICH
        SKODE = ' '
        MAT   = ' '
        DO 200 I = 1, NSIZE
          MAT(I:I) = P
          SKODE(I:I) = Q
  200   CONTINUE
C
C..MODEL FINE MESH MATERIAL PROPERTIES
C
        CALL MAKKOD(MAT, SKODE,ISEAM,0)
C
  290   IREC = LEN(MAT)
        S = CHAR(ISEAM+16)
        TEMP = 'SEAM'//S
        OPEN (UNIT = 7, FILE = TEMP, RECL = IREC)
        REWIND 7
        WRITE(7,10000)MAT
10000 FORMAT(A)
        CLOSE (7)
C
C..MODEL FINE MESH MINING STATUS CODES
C
        IREC = LEN(SKODE)
```

```
      TEMP = 'KODES'//S
      OPEN (UNIT = 7, FILE = TEMP, RECL = IREC)
      REWIND 7
      WRITE(7,10000) SKODE
      CLOSE (7)
C
```

SUBROUTINE MODSEM 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587 86/02/06. 14.17.54 PAGE 2

```
C.. TEST TO SEE IF ANY COARSE MESH EXISTS
C
      IF(NX .EQ. IFXE-IFXS+1 .AND.
     #   NY .EQ. IFYE-IFYS+1) GO TO 310
C
C
C..MODEL COARSE MESH MATERIAL PROPERTIES
C
      DO 300 I = 1,NX
        K=(I-1)*NY
        DO 300 J=1,NY
          L=K+J
          MAT(L:L) = P
          SKODE(L:L) = '0'
          IF(GOB(IGOBIN:IGOBIN) .EQ. '0') SKODE(L:L) = P
          IF(P .EQ. '1') SKODE(L:L) = ' '
          IF(P .EQ. '2') SKODE(L:L) = '$'
          IF(I .GE. IFXS .AND. I .LE. IFXE .AND.
     #       J .GE. IFYS .AND. J .LE. IFYE) THEN
              MAT(L:L) = '*'
              SKODE(L:L) = '*'
          END IF
  300 CONTINUE
      CALL IOWAIT(25)
      CALL MAKCRM(MAT, SKODE, ISEAM, 1)
      IREC = LEN(MAT)
      TEMP = 'CRCOD'//S
      OPEN (UNIT = 7, FILE = TEMP, RECL = IREC)
```

```
      REWIND 7
      WRITE(7,10000) MAT
      CLOSE (7)
C
C..MODEL COARSE MESH EXTRACTION RATIOS
C
      MAT = SKODE
      CALL IOWAIT(25)
      CALL MAKEXT(MAT, SKODE, ISEAM, 1)
      IREC = LEN(MAT)
      TEMP = 'EXRAT'//S
      OPEN (UNIT = 7, FILE = TEMP, RECL = IREC)
      REWIND 7
      WRITE(7,10000) MAT
      CLOSE (7)
  310 CALL LINEF
      CALL ANMODE
      CALL IOWAIT(25)
      PRINT 315,ISEAM
  315 FORMAT('  IF SEAM ',I5,/
     #'  IS NOT O.K.'/
     #'  YOU MAY REMODEL'/
     #'  THE ENTIRE MESH'/
     #'  FOR THIS SEAM.'/)
      CALL LINEF
      CALL ANMODE
      CALL IOWAIT(25)
  320 PRINT *,'  IS SEAM ',ISEAM,' O.K.(Y/N)'
      CALL RDTERM(1,A,I,R)
```



```
      P = A(1:1)
      IF(P .NE. 'Y' .AND. P .NE. 'N') GO TO 320
      IF (P .NE. 'Y') GO TO 100
  400 CONTINUE
     RETURN
```

```
    118                    END
--VARIABLE MAP--(LO=A)
 -NAME---ADDRESS --BLOCK----------TYPE-----------NAME---ADDRESS --BLOCK---------TYPE----
  A            732B                CHAR*1          MAT            1  DUMMY-ARG      CHAR*(*)
  CR             0B  /CHARS/       CHAR*28         NCNTRL        4B  /INTRS/        INTEGER
  FC           726B                CHAR*1          NMATS         5B  /INTRS/        INTEGER
  GOB            5B  /CHARS/       CHAR*28         NMATS2        6B  /INTRS/        INTEGER
  I            736B                INTEGER         NOSP          3B  /INTRS/        INTEGER
  ICONVR         0B  /CHARS/       CHAR*1          NSEAM         0B  /INTRS/        INTEGER
  IFLAG          7B  /INTRS/       INTEGER         NSIZE         3   DUMMY-ARG      INTEGER
  IFOUND        10B  /INTRS/       INTEGER         NX            1B  /INTRS/        INTEGER
  IFXE          21B  /INTRS/       INTEGER         NXEL         14B  /INTRS/        INTEGER
  IFXS          20B  /INTRS/       INTEGER         NXMAX        11B  /INTRS/        INTEGER
  IFYE          17B  /INTRS/       INTEGER         NY            2B  /INTRS/        INTEGER
  IFYS          16B  /INTRS/       INTEGER         NYEL         15B  /INTRS/        INTEGER
  IGOBIN       740B                INTEGER         NYMAX        12B  /INTRS/        INTEGER
  IREC         741B                INTEGER         P           731B                 CHAR*1
  ISEAM        734B                INTEGER         Q             0B  /CHARS/        CHAR*1
  IWHICH        13B  /INTRS/       INTEGER         R           737B                 REAL
  J            743B                INTEGER         S           733B                 CHAR*2
  K            742B                INTEGER         SKODE         2   DUMMY-ARG      CHAR*(*)
  KODE           3B  /CHARS/       CHAR*28         TEMP        730B                 CHAR*6
  L            744B                INTEGER         VR          727B                 CHAR*1
--PROCEDURES--(LO=A)
 -NAME------TYPE--------ARGS------CLASS-----      -NAME------TYPE--------ARGS------CLASS-----
  ANMODE                   0      SUBROUTINE       MAKCRM                    4      SUBROUTINE
  CHAR      CHAR*0         1      INTRINSIC        MAKEXT                    4      SUBROUTINE
  CHECK                    1      SUBROUTINE       MAKKOD                    4      SUBROUTINE
  IOWAIT                   1      SUBROUTINE       NEWPAG                    0      SUBROUTINE
  LEN       INTEGER        1      INTRINSIC        RDTERM                    4      SUBROUTINE
  LINEF                    0      SUBROUTINE
--STATEMENT LABELS--(LO=A)
 -LABEL-ADDRESS-----PROPERTIES---DEF  -LABEL-ADDRESS----PROPERTIES---DEF  -LABEL-ADDRESS----PROPERTIES---DEF
    100   16B                     15     290 *NO REFS*               38     320   366B                   111
    110 *NO REFS*                 18     300  INACTIVE  DO-TERM      78     400 INACTIVE  DO-TERM        116
    115   436B     FORMAT         19     310   350B                  99   10000   455B    FORMAT          44
```

```
200  INACTIVE   DO-TERM        32           315    457B     FORMAT        103

      SUBROUTINE MODSEM 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587  86/02/06. 14.17.54  PAGE 4
--ENTRY POINTS--(LO=A)
 -NAME---ADDRESS--ARGS---
  MODSEM      3B     3
--I/O UNITS--(LO=A)
 -NAME--- PROPERTIES-------------
  TAPE7   AUX/FMT/SEQ
--STATISTICS--
  PROGRAM-UNIT LENGTH             745B =  485
  CM LABELLED COMMON LENGTH        33B =  27
  CM STORAGE USED               61200B =  25216
  COMPILE TIME                  0.949 SECONDS

      SUBROUTINE CHECK 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587  86/02/06. 14.17.54  PAGE 1
      DO=-LONG/-OT,ARG=-COMMON/-FIXED,CS= USER/-FIXED,DB=-TB/-SB/-SL/-ER/-ID/-PMD/-ST,PL=5000
      FTN5,I=PGNMLBM,B=BGNMLBM,L=LGNMLBM,OPT=3.
C*********************************************************************
C     SUBROUTINE CHECK
C*********************************************************************
      SUBROUTINE CHECK(PP)
      CHARACTER *1 PP, CC
      CHARACTER  Q, ICONVR, CR*28, KODE*28
      COMMON /CHARS/ Q, ICONVR, CR, KODE
      COMMON /INTRS/ NSEAM, NX, NY, NOSP, NCNTRL, NMATS, NMATS2,
     #               IFLAG, IFOUND, NXMAX, NYMAX, IWHICH, NXEL, NYEL
     #               ,IFYS,IFYE,IFXS,IFXE
      IFLAG = 0
      DO 100 ICHK = 1,NMATS2
        CC = CR(ICHK:ICHK)
        IF (PP .EQ. CC) THEN
           IFLAG = 1
           GO TO 200
        END IF
  100 CONTINUE
```

```
    19             200 IWHICH = ICHK
    20                 RETURN
    21                 END
--VARIABLE MAP--(LO=A)
 -NAME---ADDRESS --BLOCK-----------------TYPE--------------NAME---ADDRESS --BLOCK-----------------TYPE---
  CC          52B                        CHAR*1            NMATS        5B  /INTRS/               INTEGER
  CR           0B  /CHARS/               CHAR*28           NMATS2       6B  /INTRS/               INTEGER
  ICHK        53B                        INTEGER           NOSP         3B  /INTRS/               INTEGER
  ICONVR       0B  /CHARS/               CHAR*1            NSEAM        0B  /INTRS/               INTEGER
  IFLAG        7B  /INTRS/               INTEGER           NX           1B  /INTRS/               INTEGER
  IFOUND      10B  /INTRS/               INTEGER           NXEL        14B  /INTRS/               INTEGER
  IFXE        21B  /INTRS/               INTEGER           NXMAX       11B  /INTRS/               INTEGER
  IFXS        20B  /INTRS/               INTEGER           NY           2B  /INTRS/               INTEGER
  IFYE        17B  /INTRS/               INTEGER           NYEL        15B  /INTRS/               INTEGER
  IFYS        16B  /INTRS/               INTEGER           NYMAX       12B  /INTRS/               INTEGER
  IWHICH      13B  /INTRS/               INTEGER           PP           1   DUMMY-ARG             CHAR*1
  KODE         3B  /CHARS/               CHAR*28           Q            0B  /CHARS/               CHAR*1
  NCNTRL       4B  /INTRS/               INTEGER
--STATEMENT LABELS--(LO=A)
 -LABEL-ADDRESS-----PROPERTIES----DEF
    100  INACTIVE   DO-TERM        18
    200     31B                    19

         SUBROUTINE CHECK 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587  86/02/06. 14.17.54  PAGE 2
--ENTRY POINTS--(LO=A)
 -NAME---ADDRESS--ARGS---
  CHECK       3B     1
--STATISTICS--
  PROGRAM-UNIT LENGTH           54B =  44
  CM LABELLED COMMON LENGTH     30B =  24
  CM STORAGE USED            60700B =  25024
  COMPILE TIME               0.146 SECONDS

         SUBROUTINE PLTPAT 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587  86/02/06. 14.17.54  PAGE 1
         DO=-LONG/-OT,ARG=-COMMON/-FIXED,CS= USER/-FIXED,DB=-TB/-SB/-SL/-ER/-ID/-PMD/-ST,PL=5000
         FTN5,I=PGNMLBM,B=BGNMLBM,L=LGNMLBM,OPT=3.
```

```
C******************************************************************
C     SUBROUTINE PLOTPATTERN
C******************************************************************
      SUBROUTINE PLTPAT(XMIN,XMAX,YMIN,YMAX,ISEAM)
      CHARACTER  Q, ICONVR, CR*28, KODE*28
      COMMON /CHARS/ Q, ICONVR, CR, KODE
      COMMON /INTRS/ NSEAM, NX, NY, NOSP, NCNTRL, NMATS, NMATS2,
     #               IFLAG, IFOUND, NXMAX, NYMAX, IWHICH, NXEL, NYEL
     #               ,IFYS,IFYE,IFXS,IFXE
      CALL NEWPAG
      CALL TWINDO(213,725,10,755)
      CALL DWINDO(XMIN,XMAX,YMIN,YMAX)
      J1 = (IFXE-IFXS+1) * 5
      I1 = (IFYE-IFYS+1)*5
      RJ1 = J1
      RI1 = I1
      XMN = MIN(XMAX,RJ1)
      YMN = MIN(YMAX,RI1)
      DO 100 I = 0,J1
        RI = I
        CALL MOVEA(RI,0.)
        CALL DRAWA(RI,RI1)
  100 CONTINUE
      DO 200 I = 0,I1
        RI = I
        CALL MOVEA(0.,RI)
        CALL DRAWA(RJ1,RI)
  200 CONTINUE
      DO 300 I = 0,J1,5
        RI = I
        RI = RI - 0.1
        RI2 = RI + 0.1
        IF(RI2.GE.XMIN .AND. RI2.LT.XMN) THEN
          CALL MOVEA(RI2,YMIN)
          CALL MOVREL(+15,-15)
          ITMP = I/5 + IFXS
```

```
             ENCODE(2,1000,IARAY)ITMP
 1000 FORMAT(I2)
             CALL AOUTST(2,IARAY)
             CALL MOVEA(RI,YMIN)
             CALL DRAWA(RI,RI1)
             CALL MOVEA(RI2,YMIN)
             CALL DRAWA(RI2,RI1)
           END IF
  300 CONTINUE
         DO 400 I = 0,I1,5
           RI = I
           RI = RI - 0.1
           RI2 = RI + 0.1
           IF(RI2.GE.YMIN .AND. RI2.LT.YMN) THEN
             CALL MOVEA(XMIN,RI2)
             CALL MOVREL(-15,+15)
             ITMP = I/5 + IFYS
             ENCODE(2,1000,IARAY)ITMP
             CALL AOUTST(2,IARAY)
```



```
             CALL MOVEA(XMIN,RI)
             CALL DRAWA(RJ1,RI)
             CALL MOVEA(XMIN,RI2)
             CALL DRAWA(RJ1,RI2)
           END IF
  400 CONTINUE
         CALL HOME
         CALL ANMODE
         CALL IOWAIT(15)
         PRINT *,'                    SEAM NO. ',ISEAM
         CALL TWINDO(213,725,10,755)
         CALL DWINDO(XMIN,XMAX,YMIN,YMAX)
         CALL MOVEA(XMIN,YMIN)
         RETURN
         END
```

```
--VARIABLE MAP--(LO=A)
 -NAME---ADDRESS --BLOCK-----------------TYPE--------------NAME---ADDRESS --BLOCK-----------------TYPE---
  CR          0B  /CHARS/                CHAR*28           NOSP         3B  /INTRS/                INTEGER
  I         351B                         INTEGER           NSEAM        0B  /INTRS/                INTEGER
  IARAY     360B                         INTEGER           NX           1B  /INTRS/                INTEGER
  ICONVR      0B  /CHARS/                CHAR*1            NXEL        14B  /INTRS/                INTEGER
  IFLAG       7B  /INTRS/                INTEGER           NXMAX       11B  /INTRS/                INTEGER
  IFOUND     10B  /INTRS/                INTEGER           NY           2B  /INTRS/                INTEGER
  IFXE       21B  /INTRS/                INTEGER           NYEL        15B  /INTRS/                INTEGER
  IFXS       20B  /INTRS/                INTEGER           NYMAX       12B  /INTRS/                INTEGER
  IFYE       17B  /INTRS/                INTEGER           Q            0B  /CHARS/                CHAR*1
  IFYS       16B  /INTRS/                INTEGER           RI         353B                         REAL
  ISEAM       5   DUMMY-ARG              INTEGER           RI1        346B                         REAL
  ITMP      357B                         INTEGER           RI2        356B                         REAL
  IWHICH     13B  /INTRS/                INTEGER           RJ1        345B                         REAL
  I1        344B                         INTEGER           XMAX         2   DUMMY-ARG              REAL
  J1        343B                         INTEGER           XMIN         1   DUMMY-ARG              REAL
  KODE        3B  /CHARS/                CHAR*28           XMN        347B                         REAL
  NCNTRL      4B  /INTRS/                INTEGER           YMAX         4   DUMMY-ARG              REAL
  NMATS       5B  /INTRS/                INTEGER           YMIN         3   DUMMY-ARG              REAL
  NMATS2      6B  /INTRS/                INTEGER           YMN        350B                         REAL
--PROCEDURES--(LO=A)
 -NAME------TYPE--------ARGS------CLASS-----      -NAME------TYPE--------ARGS------CLASS-----
  ANMODE                   0      SUBROUTINE       MIN       GENERIC      VAR       INTRINSIC
  AOUTST                   2      SUBROUTINE       MOVEA                    2       SUBROUTINE
  DRAWA                    2      SUBROUTINE       MOVREL                   2       SUBROUTINE
  DWINDO                   4      SUBROUTINE       NEWPAG                   0       SUBROUTINE
  HOME                     0      SUBROUTINE       TWINDO                   4       SUBROUTINE
  IOWAIT                   1      SUBROUTINE

        SUBROUTINE PLTPAT 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587  86/02/06. 14.17.54  PAGE 3
--STATEMENT LABELS--(LO=A)
 -LABEL-ADDRESS-----PROPERTIES----DEF
   100  INACTIVE    DO-TERM       23
   200  INACTIVE    DO-TERM       28
   300  INACTIVE    DO-TERM       45
```

```
  400  INACTIVE   DO-TERM        61
 1000    242B     FORMAT         38
--ENTRY POINTS--(LO=A)
 -NAME---ADDRESS--ARGS---
  PLTPAT      3B     5
--STATISTICS--
  PROGRAM-UNIT LENGTH              374B =  252
  CM LABELLED COMMON LENGTH         30B =  24
  CM STORAGE USED                61100B =  25152
  COMPILE TIME                   0.566 SECONDS
```

```
SUBROUTINE PRMAT 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587  86/02/06. 14.17.54  PAGE 1
DO=-LONG/-OT,ARG=-COMMON/-FIXED,CS= USER/-FIXED,DB=-TB/-SB/-SL/-ER/-ID/-PMD/-ST,PL=5000
FTN5,I=PGNMLBM,B=BGNMLBM,L=LGNMLBM,OPT=3.
C*********************************************************************
C     SUBROUTINE PRMAT
C*********************************************************************
      SUBROUTINE PRMAT(MAT, ISEAM, XMIN, XMAX, YMIN, YMAX, IMESH, IBLNK)
C
C.. IMESH = 0 (FINE MESH)
C           1 (COARSE MESH)
C.. IBLNK = 0 ( REPLACE CODE '1' WITH BLANK)
C           1 (DO NOT REPLACE CODE '1' WITH BLANK)
      CHARACTER *(*) MAT
      COMMON /INTRS/ NSEAM, NX, NY, NOSP, NCNTRL, NMATS, NMATS2,
     #               IFLAG, IFOUND, NXMAX, NYMAX, IWHICH, NXEL, NYEL
     #               ,IFYS,IFYE,IFXS,IFXE
      DIMENSION IMAT(100)
C
C *** BUILD ARRAY WITH MATERIAL/STATUS CODES IN CORRECT
C     ORDER FOR DISPLAYING
C
      NXSTRT = 1
      NYEND  = 1
      IF(IMESH .EQ. 0) THEN
        NYSTRT = NYEL
```

```
1            NXEND  = NXEL
1          ELSE
1            NYSTRT = NY
1            NXEND  = NX
1          END IF
           DO 2000 K = NYSTRT,NYEND,-1
             LCNT = NYSTRT - K
             L = 0
              DO 1000 I = NXSTRT,NXEND
                J = I * NYSTRT - LCNT
                IF(I.GT.XMIN .AND. I.LE.XMAX) THEN
1                 L = L + 1
1    C
1    C  *** ADD 32 TO THE COLLATING SEQUENCE NUMBER RETURNED BY ICHAR
1    C      IN ORDER TO SET THE CORRECT ASCII CHARACTER
1    C
1                 IMAT(L) = ICHAR(MAT(J:J)) + 32
1    C
1    C
1    C  ***     REPLACE MATERIAL PROPERTY '1'(MINED OUT) WITH BLANK
1    C
1                 IF(IMAT(L) .EQ. 49 .AND. IBLNK .EQ. 0) IMAT(L)= 32
1               END IF
      1000 CONTINUE
     C
     C *** DISPLAY MATERIAL/STATUS CODES IN THE MODEL GRID
     C
                IF(K.GT.YMIN .AND. K.LE.YMAX) THEN
1                 RK = K - .97
1                 RX = XMIN + .3
1                 CALL MOVEA(RX,RK)
1                 CALL ANSTR(L, IMAT)
1               END IF
```

SUBROUTINE PRMAT 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587 86/02/06. 14.17.54 PAGE 2

```
      2000 CONTINUE
```

```
      57                  RETURN
      58                  END
--VARIABLE MAP--(LO=A)
 -NAME---ADDRESS --BLOCK-----PROPERTIES-------TYPE--------------NAME---ADDRESS --BLOCK------------TYPE---
  I          310B                             INTEGER           NMATS2      6B  /INTRS/           INTEGER
  IBLNK        8   DUMMY-ARG                  INTEGER           NOSP        3B  /INTRS/           INTEGER
  IFLAG       7B   /INTRS/                    INTEGER           NSEAM       0B  /INTRS/           INTEGER
  IFOUND     10B   /INTRS/                    INTEGER           NX          1B  /INTRS/           INTEGER
  IFXE       21B   /INTRS/                    INTEGER           NXEL       14B  /INTRS/           INTEGER
  IFXS       20B   /INTRS/                    INTEGER           NXEND     303B                    INTEGER
  IFYE       17B   /INTRS/                    INTEGER           NXMAX      11B  /INTRS/           INTEGER
  IFYS       16B   /INTRS/                    INTEGER           NXSTRT    300B                    INTEGER
  IMAT       134B                             INTEGER           NY          2B  /INTRS/           INTEGER
  IMESH        7   DUMMY-ARG                  INTEGER           NYEL       15B  /INTRS/           INTEGER
  ISEAM        2   DUMMY-ARG UNUSED/*S*       INTEGER           NYEND     301B                    INTEGER
  IWHICH     13B   /INTRS/                    INTEGER           NYMAX      12B  /INTRS/           INTEGER
  J          312B                             INTEGER           NYSTRT    302B                    INTEGER
  K          304B                             INTEGER           RK        313B                    REAL
  L          307B                             INTEGER           RX        314B                    REAL
  LCNT       306B                             INTEGER           XMAX        4   DUMMY-ARG         REAL
  MAT          1   DUMMY-ARG                  CHAR*(*)          XMIN        3   DUMMY-ARG         REAL
  NCNTRL      4B   /INTRS/                    INTEGER           YMAX        6   DUMMY-ARG         REAL
  NMATS       5B   /INTRS/                    INTEGER           YMIN        5   DUMMY-ARG         REAL
--PROCEDURES--(LO=A)
 -NAME------TYPE--------ARGS------CLASS-----
  ANSTR                    2      SUBROUTINE
  ICHAR     INTEGER        1      INTRINSIC
  MOVEA                    2      SUBROUTINE
--STATEMENT LABELS--(LO=A)
 -LABEL-ADDRESS-----PROPERTIES----DEF
   1000  INACTIVE   DO-TERM        46
   2000  INACTIVE   DO-TERM        56
--ENTRY POINTS--(LO=A)
 -NAME---ADDRESS--ARGS---
  PRMAT       3B     8
```

```
--STATISTICS--
  PROGRAM-UNIT LENGTH             315B =  205
  CM LABELLED COMMON LENGTH        22B =  18
  CM STORAGE USED               61000B =  25088
  COMPILE TIME                  0.366 SECONDS
```

SUBROUTINE ENTZON 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587 86/02/06. 14.17.54 PAGE 1

```
DO=-LONG/-OT,ARG=-COMMON/-FIXED,CS= USER/-FIXED,DB=-TB/-SB/-SL/-ER/-ID/-PMD/-ST,PL=5000
FTN5,I=PGNMLBM,B=BGNMLBM,L=LGNMLBM,OPT=3.
C**********************************************************************
C     SUBROUTINE ENTERZONES
C**********************************************************************
      SUBROUTINE ENTZON(MAT,P,SKODE,FC,VR,IMESH,XMIN,XMAX,YMIN,YMAX
     #                  ,ICHNG)
      CHARACTER *(*) MAT, SKODE, P, FC, VR
      CHARACTER  Q, ICONVR, CR*28, KODE*28, GOB*28
      COMMON /CHARS/ Q, ICONVR, CR, KODE, GOB
      COMMON /INTRS/ NSEAM, NX, NY, NOSP, NCNTRL, NMATS, NMATS2,
     #               IFLAG, IFOUND, NXMAX, NYMAX, IWHICH, NXEL, NYEL
      CALL LINEF
      CALL ANMODE
      CALL IOWAIT(15)
      PRINT 60
   60 FORMAT(', MOVE CURSOR TO'/
     #' THE TWO OPPOSITE'/
     #' CORNERS OF THE' /
     #' ZONE TO CHANGE'/
     #' ENTER "X"<RETURN>'/
     #' AT EACH CORNER' /
     #' ENTER "E"<RETURN>'/
     #' WHEN DONE'/
     #' IF USING A 4054'/
     #' TERMINAL, DO NOT'/
     #' PRESS <RETURN> AFTER'/
     #' ENTERING "X" OR "E"'/)
  100 CALL ANMODE
      CALL VCURSR(IUSR,X1,Y1)
```

```
C
C..IF USER ENTERED 'E' EXIT ROUTINE
C
      IF(IUSR .EQ. 69) GO TO 999
      X1 = INT(X1) + 0.5
      Y1 = INT(Y1) + 0.5
      IF (X1 .LT. XMIN) X1 = XMIN + 0.5
      IF (X1 .GT. XMAX) X1 = XMAX - 0.5
      IF (Y1 .LT. YMIN) Y1 = YMIN + 0.5
      IF (Y1 .GT. YMAX) Y1 = YMAX - 0.5
      CALL MOVEA(X1,Y1)
      CALL MOVREL(-3,-7)
      CALL ANCHO(42)
      CALL ANMODE
      CALL VCURSR(IUSR,X2,Y2)
C
C..IF USER ENTERED 'E' EXIT ROUTINE
C
      IF (IUSR .EQ. 69) GO TO 999
      X2 = INT(X2) + 0.5
      Y2 = INT(Y2) + 0.5
      IF (X2 .LT. XMIN) X2 = XMIN + 0.5
      IF (X2 .GT. XMAX) X2 = XMAX - 0.5
      IF (Y2 .LT. YMIN) Y2 = YMIN + 0.5
      IF (Y2 .GT. YMAX) Y2 = YMAX - 0.5
      X1 = INT(X1) + 1
      X2 = INT(X2) + 1
```

SUBROUTINE ENTZON 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587 86/02/06. 14.17.54 PAGE 2

```
      Y1 = INT(Y1) + 1
      Y2 = INT(Y2) + 1
      T1 = X1
      T2 = X2
      X1 = MIN(T1,T2)
      X2 = MAX(T1,T2)
      T1 = Y1
```

```
      T2 = Y2
      Y1 = MIN(T1,T2)
      Y2 = MAX(T1,T2)
      NYDX = NYEL
      IF(IFLAG.GT.1) NYDX=NY
      DO 300 J = X1,X2
        K = (J-1) * NYDX
        DO 200 I = Y1,Y2
          L = K + I
C
C.. FINE MESH CHANGES
C
          IF( IMESH .EQ. 0) THEN
            MAT(L:L) = P
            SKODE(L:L) = Q
            GO TO 200
          END IF
C
C.. COARSE MESH CHANGES
C
          IF( MAT(L:L) .EQ. '*') GO TO 200
          IF( ICHNG .GT. 0) GO TO 125
C
C.. COARSE MESH MATERIAL PROPERTIES CHANGES
C
          MAT(L:L) = P
          IF( P .EQ. '1') THEN
            SKODE(L:L) = ' '
          ELSE IF( P .EQ. '2') THEN
                  SKODE(L:L) = '$'
               ELSE IF(GOB(IWHICH:IWHICH) .EQ. '0') THEN
                     SKODE(L:L) = P
                    ELSE
                     SKODE(L:L) = '0'
          END IF
          GO TO 200
```

```
C
C..  COARSH MESH EXTRACTION RATIO CODES CHANGES
C
  125     IF(MAT(L:L) .EQ. ' ') GO TO 200
          IF(MAT(L:L) .EQ. '$') GO TO 200
          DO 150 N=3,28
            IF(MAT(L:L) .EQ. CR(N:N)) GO TO 200
  150     CONTINUE
          MAT(L:L) = P
  200   CONTINUE
  300 CONTINUE
C
      X1=X1-1
      Y1=Y1-1
```

SUBROUTINE ENTZON 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587 86/02/06. 14.17.54 PAGE 3

```
      CALL PRMAT(MAT,ISEAM,X1,X2,Y1,Y2,IMESH, 1)
C
      GO TO 100
  999 RETURN
      END
```

```
--VARIABLE MAP--(LO=A)
 -NAME---ADDRESS --BLOCK-----PROPERTIES--TYPE--------------NAME---ADDRESS --BLOCK-----PROPERTIES--TYPE---
  CR          0B  /CHARS/               CHAR*28            NSEAM       0B  /INTRS/                 INTEGER
  FC           4  DUMMY-ARG UNUSED      CHAR*(*)           NX          1B  /INTRS/                 INTEGER
  GOB         5B  /CHARS/               CHAR*28            NXEL       14B  /INTRS/                 INTEGER
  I         554B                        INTEGER            NXMAX      11B  /INTRS/                 INTEGER
  ICHNG       11  DUMMY-ARG             INTEGER            NY          2B  /INTRS/                 INTEGER
  ICONVR      0B  /CHARS/               CHAR*1             NYDX      550B                          INTEGER
  IFLAG       7B  /INTRS/               INTEGER            NYEL       15B  /INTRS/                 INTEGER
  IFOUND     10B  /INTRS/               INTEGER            NYMAX      12B  /INTRS/                 INTEGER
  IMESH        6  DUMMY-ARG             INTEGER            P           2   DUMMY-ARG               CHAR*(*)
  ISEAM     560B            *S*         INTEGER            Q           0B  /CHARS/                 CHAR*1
  IUSR      543B                        INTEGER            SKODE       3   DUMMY-ARG               CHAR*(*)
  IWHICH     13B  /INTRS/               INTEGER            T1        NONE                          REAL
  J         551B                        INTEGER            T2        NONE                          REAL
```

```
  K           553B                           INTEGER          VR           5   DUMMY-ARG UNUSED      CHAR*(*)
  KODE          3B  /CHARS/                  CHAR*28          XMAX         8   DUMMY-ARG             REAL
  L           556B                           INTEGER          XMIN         7   DUMMY-ARG             REAL
  MAT           1   DUMMY-ARG                CHAR*(*)         X1         544B                        REAL
  N           557B                           INTEGER          X2         546B                        REAL
  NCNTRL        4B  /INTRS/                  INTEGER          YMAX        10   DUMMY-ARG             REAL
  NMATS         5B  /INTRS/                  INTEGER          YMIN         9   DUMMY-ARG             REAL
  NMATS2        6B  /INTRS/                  INTEGER          Y1         545B                        REAL
  NOSP          3B  /INTRS/                  INTEGER          Y2         547B                        REAL
--PROCEDURES--(LO=A)
 -NAME------TYPE--------ARGS------CLASS-----       NAME------TYPE--------ARGS------CLASS-----
  ANCHO                    1      SUBROUTINE       MIN       GENERIC     VAR       INTRINSIC
  ANMODE                   0      SUBROUTINE       MOVEA                    2      SUBROUTINE
  INT       GENERIC        1      INTRINSIC        MOVREL                   2      SUBROUTINE
  IOWAIT                   1      SUBROUTINE       PRMAT                    8      SUBROUTINE
  LINEF                    0      SUBROUTINE       VCURSR                   3      SUBROUTINE
  MAX       GENERIC     VAR       INTRINSIC
--STATEMENT LABELS--(LO=A)
 -LABEL-ADDRESS-----PROPERTIES----DEF       -LABEL-ADDRESS-----PROPERTIES----DEF
     60    355B     FORMAT         15           200     313B     DO-TERM       108
    100     26B                    27           300  INACTIVE    DO-TERM       109
    125    261B                   102           999     334B                   116
    150  INACTIVE   DO-TERM       106
       SUBROUTINE ENTZON 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587  86/02/06. 14.17.54  PAGE 4
--ENTRY POINTS--(LO=A)
 -NAME---ADDRESS--ARGS---
  ENTZON      3B    11
--STATISTICS--
  PROGRAM-UNIT LENGTH            563B =  371
  CM LABELLED COMMON LENGTH       27B =  23
  CM STORAGE USED              61200B =  25216
  COMPILE TIME                 0.982 SECONDS

       SUBROUTINE MAKKOD 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587  86/02/06. 14.17.54  PAGE 1
       DO=-LONG/-OT,ARG=-COMMON/-FIXED,CS= USER/-FIXED,DB=-TB/-SB/-SL/-ER/-ID/-PMD/-ST,PL=5000
       FTN5,I=PGNMLBM,B=BGNMLBM,L=LGNMLBM,OPT=3.
```

```
C**************************************************************
C     SUBROUTINE MAKEKODES
C**************************************************************
      SUBROUTINE MAKKOD(MAT, SKODE, ISEAM, IMESH)
      CHARACTER A, VR, MAT*(*), FC
      CHARACTER Q, ICONVR, CR*28, KODE*28
      COMMON /CHARS/ Q, ICONVR, CR, KODE
      COMMON /INTRS/ NSEAM, NX, NY, NOSP, NCNTRL, NMATS, NMATS2,
     #               IFLAG, IFOUND, NXMAX, NYMAX, IWHICH, NXEL, NYEL
     #               ,IFYS,IFYE,IFXS,IFXE
        IBLNK = 0
        ICHNG = 0
        XMIN = 0
        XMAX = 60
        YMIN = 0
        YMAX = 60
        XSTRB = 1
        YSTRB = 1
        CALL ANMODE
        CALL IOWAIT(15)
  220   IF (IFXE-IFXS .GT. 11) THEN
           CALL ANMODE
           PRINT 10010
10010 FORMAT('  FINE MESH AREA'/
     #       '  X-AXIS IS GREATER'/
     #       '  THAN CAN BE '/
     #       '  DISPLAYED ON SCREEN'/
     #       '  ENTER STARTING'/
     #       '  X-AXIS BLOCK'/)
          CALL RDTERM(3,T,I,XSTRB)
          IF( XSTRB .LT. IFXS .OR. XSTRB .GT. IFXE) THEN
            PRINT *,'STARTING BLOCK MUST BE ',IFXS,'THRU ',IFXE
            GO TO 220
          END IF
          XSTRB = XSTRB - IFXS + 1
        END IF
```

```
        224   IF (IFYE-IFYS .GT. 11) THEN
                CALL IOWAIT(15)
                CALL ANMODE
                PRINT 10020
      10020 FORMAT('  FINE MESH AREA'/
          #        '  Y-AXIS IS GREATER'/
          #        '  THAN CAN BE '/
          #        '  DISPLAYED ON SCREEN'/
          #        '  ENTER STARTING'/
          #        '  Y-AXIS BLOCK'/)
                CALL RDTERM(3,T,I,YSTRB)
                IF( YSTRB .LT. IFYS .OR. YSTRB .GT. IFYE) THEN
                  PRINT *,'STARTING BLOCK MUST BE ',IFYS,'THRU ',IFYE
                  GO TO 224
                END IF
                YSTRB = YSTRB -IFYS + 1
              END IF
        228   XMIN = (XSTRB - 1) * 5
              XMAX = XMIN + 60

SUBROUTINE MAKKOD 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587  86/02/06. 14.17.54  PAGE 2
              YMIN = (YSTRB - 1) * 5
              YMAX = YMIN + 60
              CALL PLTPAT(XMIN,XMAX,YMIN,YMAX,ISEAM)
              CALL PRMAT(MAT, ISEAM, XMIN, XMAX, YMIN, YMAX, IMESH, IBLNK)
        230   CALL HOME
              CALL LINEF
              CALL ANMODE
              PRINT *,' CHANGES(Y/N)'
              CALL RDTERM(1,A,I,R)
              IF(A .NE. 'Y' .AND. A .NE. 'N') GO TO 230
        250   IF (A .EQ. 'Y') THEN
                CALL CHANGE(MAT, FC, VR, ISEAM,XMIN,XMAX,YMIN,YMAX,
          #               IMESH, ICHNG, SKODE)
                GO TO 230
              END IF
```

```
            255   PRINT *, ' DO YOU WANT'
                  PRINT *, ' A HARD COPY(Y/N)'
                  CALL RDTERM(1,A,I,R)
                  IF(A .NE. 'Y' .AND. A .NE. 'N') GO TO 255
                  IF(A .EQ. 'Y') THEN
                    CALL IOWAIT(15)
                    CALL HDCOPY
                    CALL IOWAIT(25)
                  END IF
                  IF(IFYE-IFYS .LE. 11 .AND.
                #     IFXE-IFXS .LE. 11) GO TO 999
            260   CALL ANMODE
                  PRINT 265
            265   FORMAT('DO YOU WANT TO'/
                #        'CHANGE OR EXAMINE'/
                #        'OTHER PARTS OF THE'/
                #        'MODEL NOT VISIBLE'/
                #        'ON THE SCREEN(Y/N)')
                  CALL RDTERM(1,A,I,R)
                  IF(A .NE. 'Y' .AND. A .NE. 'N') GO TO 260
            280   IF (A .EQ. 'Y') GO TO 220
            999   RETURN
                  END
--VARIABLE MAP--(LO=A)
 -NAME---ADDRESS --BLOCK-----------------TYPE---------------NAME---ADDRESS --BLOCK-----------------TYPE---
  A         450B                         CHAR*1             ISEAM         3  DUMMY-ARG               INTEGER
  CR          0B  /CHARS/                CHAR*28            IWHICH      13B  /INTRS/                 INTEGER
  FC        452B                         CHAR*1             KODE         3B  /CHARS/                 CHAR*28
  I         464B                         INTEGER            MAT           1  DUMMY-ARG               CHAR*(*)
  IBLNK     453B                         INTEGER            NCNTRL       4B  /INTRS/                 INTEGER
  ICHNG     454B                         INTEGER            NMATS        5B  /INTRS/                 INTEGER
  ICONVR      0B  /CHARS/                CHAR*1             NMATS2       6B  /INTRS/                 INTEGER
  IFLAG       7B  /INTRS/                INTEGER            NOSP         3B  /INTRS/                 INTEGER
  IFOUND     10B  /INTRS/                INTEGER            NSEAM        0B  /INTRS/                 INTEGER
  IFXE       21B  /INTRS/                INTEGER            NX           1B  /INTRS/                 INTEGER
  IFXS       20B  /INTRS/                INTEGER            NXEL        14B  /INTRS/                 INTEGER
```

```
 IFYE        17B  /INTRS/                INTEGER              NXMAX       11B  /INTRS/           INTEGER
 IFYS        16B  /INTRS/                INTEGER              NY           2B  /INTRS/           INTEGER
 IMESH        4   DUMMY-ARG              INTEGER              NYEL        15B  /INTRS/           INTEGER
       SUBROUTINE MAKKOD 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587  86/02/06. 14.17.54  PAGE 3
 -NAME---ADDRESS --BLOCK-----      ------TYPE-----            -NAME---ADDRESS --BLOCK----       -TYPE---
 NYMAX       12B  /INTRS/                INTEGER              XMAX       456B                    REAL
 Q            0B  /CHARS/                CHAR*1               XMIN       455B                    REAL
 R          465B                         REAL                 XSTRB      461B                    REAL
 SKODE        2   DUMMY-ARG              REAL                 YMAX       460B                    REAL
 T          463B                         REAL                 YMIN       457B                    REAL
 VR         451B                         CHAR*1               YSTRB      462B                    REAL
--PROCEDURES--(LO=A)
 -NAME------TYPE--------ARGS------CLASS-----      -NAME------TYPE--------ARGS------CLASS-----
 ANMODE                    0      SUBROUTINE       LINEF                    0      SUBROUTINE
 CHANGE                   11      SUBROUTINE       PLTPAT                   5      SUBROUTINE
 HDCOPY                    0      SUBROUTINE       PRMAT                    8      SUBROUTINE
 HOME                      0      SUBROUTINE       RDTERM                   4      SUBROUTINE
 IOWAIT                    1      SUBROUTINE
--STATEMENT LABELS--(LO=A)
 -LABEL-ADDRESS-------DEF      -LABEL-ADDRESS-----PROPERTIES----DEF     -LABEL-ADDRESS-----PROPERTIES----DEF
   220     22B         21         250 *NO REFS*                  66        280 *NO REFS*                  91
   224     46B         37         255    147B                    71        999    221B                    92
   228 *NO REFS*       54         260    202B                    82      10010    250B     FORMAT         24
   230    114B         60         265    304B     FORMAT         84      10020    266B     FORMAT         41
--ENTRY POINTS--(LO=A)
 -NAME---ADDRESS--ARGS---
 MAKKOD      3B     4
--STATISTICS--
 PROGRAM-UNIT LENGTH              474B =  316
 CM LABELLED COMMON LENGTH         30B =  24
 CM STORAGE USED                61200B =  25216
 COMPILE TIME                   0.635 SECONDS

       SUBROUTINE CHANGE 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587  86/02/06. 14.17.54  PAGE 1
       DO=-LONG/-OT,ARG=-COMMON/-FIXED,CS= USER/-FIXED,DB=-TB/-SB/-SL/-ER/-ID/-PMD/-ST,PL=5000
       FTN5,I=PGNMLBM,B=BGNMLBM,L=LGNMLBM,OPT=3.
```

```
      SUBROUTINE CHANGE(MAT, FC, VR, ISEAM, XMIN,XMAX,
     #                 YMIN,YMAX,IMESH, ICHNG, SKODE)
      CHARACTER *(*) MAT, FC, VR, SKODE
      CHARACTER  Q, ICONVR, CR*28, KODE*28, P
      COMMON /CHARS/ Q, ICONVR, CR, KODE
      COMMON /INTRS/ NSEAM, NX, NY, NOSP, NCNTRL, NMATS, NMATS2,
     #               IFLAG, IFOUND, NXMAX, NYMAX, IWHICH, NXEL, NYEL
     #               ,IFYS,IFYE,IFXS,IFXE,IEX
  100 CALL HOME
      CALL MOVREL(0,-100)
  150 CALL IOWAIT(15)
      CALL ANMODE
      IFLAG = 0
C.. IF EXTRACTION RATIO CODES TO BE CHANGED
C
      IF(ICHNG .GT. 0) GO TO 600
C
C..CHANGE MATERIAL PROPERTIES
C
  200 PRINT 2000
 2000 FORMAT(//' 1 = OPENING;'/
     #' 2 = RIGID;' /
     #' A-Z = MATERIALS;'/
     #'  ENTER LETTER'/
     #'  OR NUMBER'/
     #'  TO SUBSTITUTE'/)
      CALL RDTERM(1,P,I,R)
10000 FORMAT(A)
      CALL CHECK(P)
      IF (IFLAG .EQ. 0) GO TO 200
      Q = KODE(IWHICH:IWHICH)
C
C.. IF COARSE MESH
C
      ITAG = 3
```

```
              IBLNK = 0
              IF(IMESH .GT. 0) GO TO 700
        C
          500 CALL ENTZON(MAT, P, SKODE, FC, VR, IMESH, XMIN, XMAX, YMIN, YMAX
             #            ,ICHNG)
              CALL PLTPAT(XMIN,XMAX,YMIN,YMAX,ISEAM)
              CALL PRMAT(MAT, ISEAM, XMIN, XMAX, YMIN, YMAX, IMESH, 0)
              GO TO 999
        C
        C..CHANGE EXTRACTION RATIO CODES
        C
          600 PRINT 2200, IEX
         2200 FORMAT( /'CHANGE EXTRACTION CODES'/
             #        /'   VALID CODES 0 THRU ',I2,/)
              CALL RDTERM(2,P,IC,R)
              IF( IC .LT. 0 .OR. IC .GT. IEX) THEN
                PRINT *,'INVALID EXTRACTION CODE; REENTER'
                GO TO 600
              END IF
```

SUBROUTINE CHANGE 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587 86/02/06. 14.17.54 PAGE 2

```
              WRITE(P,'(I1)') IC
              ITAG = 4
              IBLNK = 1
          700 IFLAG = ITAG
              CALL ENTZON(MAT,P,SKODE,FC,VR, IMESH, XMIN, XMAX, YMIN, YMAX
             #            ,ICHNG)
              CALL PLTCRS(NX,NY,ISEAM,ITAG)
              CALL PRMAT(MAT, ISEAM, XMIN, XMAX, YMIN, YMAX, IMESH, IBLNK)
          999 RETURN
              END
--VARIABLE MAP--(LO=A)
 -NAME---ADDRESS --BLOCK-----PROPERTIES--TYPE---------SIZE NAME---ADDRESS --BLOCK-----PROPERTIES--TYPE---
  CR          0B  /CHARS/                 CHAR*28          NCNTRL       4B  /INTRS/                 INTEGER
  FC           2  DUMMY-ARG               CHAR*(*)         NMATS        5B  /INTRS/                 INTEGER
  I         363B           *S*            INTEGER          NMATS2       6B  /INTRS/                 INTEGER
```

```
 IBLNK       366B                       INTEGER         NOSP         3B  /INTRS/            INTEGER
 IC          367B                       INTEGER         NSEAM        0B  /INTRS/            INTEGER
 ICHNG        10   DUMMY-ARG            INTEGER         NX           1B  /INTRS/            INTEGER
 ICONVR       0B   /CHARS/              CHAR*1          NXEL        14B  /INTRS/            INTEGER
 IEX         22B   /INTRS/              INTEGER         NXMAX       11B  /INTRS/            INTEGER
 IFLAG        7B   /INTRS/              INTEGER         NY           2B  /INTRS/            INTEGER
 IFOUND      10B   /INTRS/              INTEGER         NYEL        15B  /INTRS/            INTEGER
 IFXE        21B   /INTRS/              INTEGER         NYMAX       12B  /INTRS/            INTEGER
 IFXS        20B   /INTRS/              INTEGER         P          362B                     CHAR*1
 IFYE        17B   /INTRS/              INTEGER         Q            0B  /CHARS/            CHAR*1
 IFYS        16B   /INTRS/              INTEGER         R          364B                     REAL
 IMESH        9    DUMMY-ARG            INTEGER         SKODE       11   DUMMY-ARG          CHAR*(*)
 ISEAM        4    DUMMY-ARG            INTEGER         VR           3   DUMMY-ARG          CHAR*(*)
 ITAG        365B                       INTEGER         XMAX         6   DUMMY-ARG          REAL
 IWHICH      13B   /INTRS/              INTEGER         XMIN         5   DUMMY-ARG          REAL
 KODE         3B   /CHARS/              CHAR*28         YMAX         8   DUMMY-ARG          REAL
 MAT          1    DUMMY-ARG            CHAR*(*)        YMIN         7   DUMMY-ARG          REAL
--PROCEDURES--(LO=A)
 -NAME------TYPE--------ARGS------CLASS-----       -NAME------TYPE--------ARGS------CLASS-----
 ANMODE                     0     SUBROUTINE        MOVREL                    2       SUBROUTINE
 CHECK                      1     SUBROUTINE        PLTCRS                    4       SUBROUTINE
 ENTZON                    11     SUBROUTINE        PLTPAT                    5       SUBROUTINE
 HOME                       0     SUBROUTINE        PRMAT                     8       SUBROUTINE
 IOWAIT                     1     SUBROUTINE        RDTERM                    4       SUBROUTINE
--STATEMENT LABELS--(LO=A)
 -LABEL-ADDRESS----------DEF       -LABEL-ADDRESS----------DEF       -LABEL-ADDRESS-----PROPERTIES----DEF
   100 *NO REFS*          10          600     110B           48        2000     207B     FORMAT         22
   150 *NO REFS*          12          700     126B           59        2200     225B     FORMAT         49
   200     20B            21          999     172B           64       10000     223B     FORMAT         29
   500 *NO REFS*          40

      SUBROUTINE CHANGE 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587  86/02/06. 14.17.54  PAGE 3
--ENTRY POINTS--(LO=A)
 -NAME---ADDRESS--ARGS---
 CHANGE      3B     11
```

```
--STATISTICS--
  PROGRAM-UNIT LENGTH            370B =  248
  CM LABELLED COMMON LENGTH       31B =  25
  CM STORAGE USED              61100B =  25152
  COMPILE TIME                 0.497 SECONDS
```

```
SUBROUTINE PUTTGR 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587  86/02/06. 14.17.54  PAGE 1
DO=-LONG/-OT,ARG=-COMMON/-FIXED,CS= USER/-FIXED,DB=-TB/-SB/-SL/-ER/-ID/-PMD/-ST,PL=5000
FTN5,I=PGNMLBM,B=BGNMLBM,L=LGNMLBM,OPT=3.
C*********************************************************************
C     SUBROUTINE PUTTGR
C*********************************************************************
      SUBROUTINE PUTTGR(NBLOCK, MAT)
      CHARACTER  Q, ICONVR, CR*28, KODE*28
      COMMON /CHARS/ Q, ICONVR, CR, KODE
      COMMON /INTRS/ NSEAM, NX, NY, NOSP, NCNTRL, NMATS, NMATS2,
     #               IFLAG, IFOUND, NXMAX, NYMAX, IWHICH, NXEL, NYEL
      CHARACTER A*80, F*6, MAT*(*), FNAME*6
      CALL NEWPAG
      IREC = LEN(MAT)
      CALL IOWAIT(15)
      PRINT *,' ENTER FILE NAME(FOR FILE TO BE CREATED)'
      CALL RDTERM(1,FNAME,I,R)
      OPEN(UNIT=2, FILE=FNAME)
      OPEN(UNIT=1, FILE= 'CNTRLS', STATUS='OLD')
      PRINT *,'...NOW WRITING CONTROL CARDS'
      REWIND 1
      DO 200 I = 1,NCNTRL
        READ (1,10000)A
        WRITE(2,10000)A
  200 CONTINUE
  210 CLOSE (1, STATUS = 'DELETE')
      DO 300 I = 1, NSEAM
        F = 'SEAM'//CHAR(I+16)
        OPEN(UNIT=1,FILE=F, STATUS='OLD', RECL = IREC)
        PRINT *,'...NOW WRITING MATERIAL CODES FOR SEAM ',I
```

```
            REWIND 1
            READ(1,10000)MAT
            CALL WRTMAT(MAT,0)
            CLOSE (1, STATUS = 'DELETE')
      300 CONTINUE
          DO 400 I = 1, NSEAM
            F = 'KODES'//CHAR(I+16)
            OPEN(UNIT=1,FILE=F, STATUS='OLD', RECL = IREC)
            REWIND 1
            READ(1,10000)MAT
            PRINT *,'...NOW WRITING STATUS CODES FOR SEAM ',I
            CALL WRTMAT(MAT,0)
            CLOSE (1, STATUS = 'DELETE')
      400 CONTINUE
C
C.. CHECK TO SEE IF ANY COARSE MESH EXISTS
C
          IF(NX .EQ. IFXE-IFXS+1 .AND.
         #   NY .EQ. IFYE-IFYS+1) GO TO 480
C
C
C..WRITE COARSE MESH MATERIAL CODES
C
          DO 420 I = 1, NSEAM
            F = 'CRCOD'//CHAR(I+16)
            OPEN(UNIT=1,FILE=F, STATUS='OLD', RECL = IREC)
            REWIND 1
            READ(1,10000) MAT

SUBROUTINE PUTTGR 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587  86/02/06. 14.17.54  PAGE 2
            PRINT *,'...NOW WRITING COARSE MESH MATERIAL CODES FOR SEAM ',I
            CALL WRTMAT(MAT,1)
            CLOSE (1, STATUS = 'DELETE')
      420 CONTINUE
C
C..WRITE COARSE MESH EXTRACTION RATIOS
```

```
C
      DO 440 I = 1, NSEAM
        F = 'EXRAT'//CHAR(I+16)
        OPEN(UNIT=1,FILE=F, STATUS='OLD', RECL = IREC)
        REWIND 1
        READ(1,10000) MAT
        PRINT *,'...NOW WRITING COARSE MESH EXTRACTION CODES'
        PRINT *,'   FOR SEAM ',I
        DO 410 J=1,NBLOCK
          IF(MAT(J:J) .EQ. ' ') MAT(J:J) = '0'
          IF(MAT(J:J) .EQ. '$') MAT(J:J) = '0'
          DO 405 JI=3,28
            IF(MAT(J:J) .EQ. CR(JI:JI) ) MAT(J:J) = '0'
  405     CONTINUE
  410   CONTINUE
        CALL WRTMAT(MAT,1)
        CLOSE (1, STATUS = 'DELETE')
  440 CONTINUE
  480 IF (NOSP .GT. 0) THEN
        OPEN (UNIT=3,FILE='OFSM', STATUS='OLD')
        PRINT *,'...NOW WRITING OFF-SEAM CONTROL CARDS'
        REWIND 3
  500   READ(3,10000,END=510) A
        WRITE(2,10000)A
        GO TO 500
  510   CLOSE (3, STATUS = 'DELETE')
      END IF
      REWIND 2

      F = FNAME
  700 IERR = 0
      CALL PF('SAVE',FNAME,F,'RC',IERR)
  800 IF(IERR .GT. 0) THEN
        PRINT *,' FILE ',F,' ALREADY EXISTS'
        PRINT *,' ENTER NEW PERMANENT FILE NAME'
        CALL RDTERM(1,F,I,R)
```

```
                    GO TO 700
                END IF
                CLOSE (2, STATUS='KEEP')
                RETURN
          10000 FORMAT(A)
                END
     SUBROUTINE PUTTGR 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587  86/02/06. 14.17.54  PAGE 3
--VARIABLE MAP--(LO=A)
 -NAME---ADDRESS --BLOCK-----PROPERTIES-------TYPE----          --NAME---ADDRESS --BLOCK-----------TYPE---
  A          724B                             CHAR*80            KODE          3B  /CHARS/          CHAR*28
  CR          0B  /CHARS/                     CHAR*28            MAT           2   DUMMY-ARG        CHAR*(*)
  F          734B                             CHAR*6             NBLOCK        1   DUMMY-ARG        INTEGER
  FNAME      735B                             CHAR*6             NCNTRL        4B  /INTRS/          INTEGER
  I          737B                             INTEGER            NMATS         5B  /INTRS/          INTEGER
  ICONVR      0B  /CHARS/                     CHAR*1             NMATS2        6B  /INTRS/          INTEGER
  IERR       754B                             INTEGER            NOSP          3B  /INTRS/          INTEGER
  IFLAG       7B  /INTRS/                     INTEGER            NSEAM         0B  /INTRS/          INTEGER
  IFOUND     10B  /INTRS/                     INTEGER            NX            1B  /INTRS/          INTEGER
  IFXE       744B             UND/*S*         INTEGER            NXEL         14B  /INTRS/          INTEGER
  IFXS       745B             UND/*S*         INTEGER            NXMAX        11B  /INTRS/          INTEGER
  IFYE       746B             UND/*S*         INTEGER            NY            2B  /INTRS/          INTEGER
  IFYS       747B             UND/*S*         INTEGER            NYEL         15B  /INTRS/          INTEGER
  IREC       736B                             INTEGER            NYMAX        12B  /INTRS/          INTEGER
  IWHICH     13B  /INTRS/                     INTEGER            Q             0B  /CHARS/          CHAR*1
  J          752B                             INTEGER            R           740B                   REAL
  JI         753B                             INTEGER
--PROCEDURES--(LO=A)
 -NAME------TYPE--------ARGS------CLASS-----       -NAME------TYPE--------ARGS------CLASS-----
  CHAR      CHAR*0         1      INTRINSIC         PF                       5      SUBROUTINE
  IOWAIT                   1      SUBROUTINE        RDTERM                   4      SUBROUTINE
  LEN       INTEGER        1      INTRINSIC         WRTMAT                   2      SUBROUTINE
  NEWPAG                   0      SUBROUTINE
--STATEMENT LABELS--(LO=A)
 -LABEL-ADDRESS----PROPERTIES--DEF   -LABEL-ADDRESS----PROPERTIES--DEF   -LABEL-ADDRESS----PROPERTIES--DEF
   200  INACTIVE  DO-TERM      22        410  INACTIVE  DO-TERM     76      510    333B                 87
   210 *NO REFS*               23        420  INACTIVE  DO-TERM     59      700    341B                 92
```

```
   300  INACTIVE   DO-TERM       32       440   INACTIVE   DO-TERM       79       800 *NO REFS*              94
   400  INACTIVE   DO-TERM       41       480     317B                   80     10000      451B     FORMAT     102
   405  INACTIVE   DO-TERM       75       500     326B                   84
--ENTRY POINTS--(LO=A)
 -NAME---ADDRESS--ARGS---
  PUTTGR      3B     2
--I/O UNITS--(LO=A)
 -NAME--- PROPERTIES-------------
  TAPE1   AUX/FMT/SEQ
  TAPE2   AUX/FMT/SEQ
  TAPE3   AUX/FMT/SEQ

         SUBROUTINE PUTTGR 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587  86/02/06. 14.17.54  PAGE 4
--STATISTICS--
  PROGRAM-UNIT LENGTH              755B =  493
  CM LABELLED COMMON LENGTH         24B =  20
  CM STORAGE USED                61200B =  25216
  COMPILE TIME                   0.955 SECONDS

         SUBROUTINE RDTERM 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587  86/02/06. 14.17.54  PAGE 1
         DO=-LONG/-OT,ARG=-COMMON/-FIXED,CS= USER/-FIXED,DB=-TB/-SB/-SL/-ER/-ID/-PMD/-ST,PL=5000
         FTN5,I=PGNMLBM,B=BGNMLBM,L=LGNMLBM,OPT=3.
      C
      C...SUBROUTINE READ TERMINAL
      C   THIS SUBROUTINE WILL READ ALL INPUT FROM
      C   THE TERMINAL.  PARAMETER ITYPE WILL DETERMINE
      C   IF THE VARIABLE TO BE READ IS:
      C      1 - CHARACTER
      C      2 - INTEGER
      C      3 - REAL
      C
            SUBROUTINE RDTERM(ITYPE,CARATR,INTAGR,REALL)
      C
            CHARACTER CARATR*(*)
      C
        100 GO TO (200, 300, 400) ITYPE
```

```
C
C..READ CHARACTER VARIABLE
C
  200 READ (5,10000,END=2000) CARATR
      RETURN
C
C..READ INTEGER VARIABLE
C
  300 READ (5,10100,END=2000) INTAGR
      RETURN
C
C..READ REAL(FLOATING POINT) VARIABLE
C
  400 READ (5,10200,END=2000) REALL
      RETURN
C
C..END OF FILE OR <CR> ENCOUNTERED
C
 2000 CLOSE(5)
      OPEN(UNIT=5,FILE='INPUT')
      PRINT *,'?'
      GO TO 100
10000 FORMAT(A)
10100 FORMAT(I10)
10200 FORMAT(F10.0)
      END
--VARIABLE MAP--(LO=A)
 -NAME---ADDRESS --BLOCK-----PROPERTIES-------TYPE---------SIZE
  CARATR      2   DUMMY-ARG                   CHAR*(*)
  INTAGR      3   DUMMY-ARG                   INTEGER
  ITYPE       1   DUMMY-ARG                   INTEGER
  REALL       4   DUMMY-ARG                   REAL

       SUBROUTINE RDTERM 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587  86/02/06. 14.17.54  PAGE 2
--STATEMENT LABELS--(LO=A)
 -LABEL-ADDRESS-----PROPERTIES----DEF       -LABEL-ADDRESS-----PROPERTIES----DEF
```

```
   100      11B                   14         2000     33B                      33
   200      16B                   18        10000     50B     FORMAT           37
   300      25B                   23        10100     52B     FORMAT           38
   400      30B                   28        10200     54B     FORMAT           39
--ENTRY POINTS--(LO=A)
 -NAME---ADDRESS--ARGS---
  RDTERM      3B     4
--I/O UNITS--(LO=A)
 -NAME--- PROPERTIES-------------
  TAPE5   AUX/FMT/SEQ
--STATISTICS--
  PROGRAM-UNIT LENGTH                112B =  74
  CM STORAGE USED                  60700B =  25024
  COMPILE TIME                     0.167 SECONDS
```

SUBROUTINE MAKCRM 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587 86/02/06. 14.17.54 PAGE 1

```
DO=-LONG/-OT,ARG=-COMMON/-FIXED,CS= USER/-FIXED,DB=-TB/-SB/-SL/-ER/-ID/-PMD/-ST,PL=5000
FTN5,I=PGNMLBM,B=BGNMLBM,L=LGNMLBM,OPT=3.
C**************************************************************
C     SUBROUTINE TO MAKE COARSE GRID MATERIAL PROPERTIES
C**************************************************************
      SUBROUTINE MAKCRM(MAT, SKODE, ISEAM, IMESH)
      CHARACTER A, VR, MAT*(*), FC, SKODE*(*)
      CHARACTER Q, ICONVR, CR*28, KODE*28
      COMMON /CHARS/ Q, ICONVR, CR, KODE
      COMMON /INTRS/ NSEAM, NX, NY, NOSP, NCNTRL, NMATS, NMATS2,
     #               IFLAG, IFOUND, NXMAX, NYMAX, IWHICH, NXEL, NYEL
     #               ,IFYS,IFYE,IFXS,IFXE
      IBLNK = 0
      ICHNG = 0
        XMIN = 0
        XMAX = 60
        YMIN = 0
        YMAX = 60
        CALL PLTCRS(NX,NY,ISEAM,3)
        CALL PRMAT(MAT, ISEAM, XMIN, XMAX, YMIN, YMAX, IMESH, IBLNK)
```

```
230   CALL HOME
      CALL LINEF
      CALL ANMODE
      PRINT *,' CHANGES(Y/N)'
      CALL RDTERM(1,A,I,R)
      IF(A .NE. 'Y' .AND. A .NE. 'N') GO TO 230
250   IF (A .EQ. 'Y') THEN
      ITAG = 3
        CALL CHANGE(MAT, FC, VR, ISEAM,XMIN,XMAX,YMIN,YMAX,
     #            IMESH, ICHNG, SKODE)
        GO TO 230
      END IF
255   PRINT *, ' DO YOU WANT'
      PRINT *, ' A HARD COPY(Y/N)'
      CALL RDTERM(1,A,I,R)
      IF(A .NE. 'Y' .AND. A .NE. 'N') GO TO 255
      IF(A .EQ. 'Y') THEN
        CALL IOWAIT(15)
        CALL HDCOPY
        CALL IOWAIT(25)
      END IF
999   RETURN
      END
--VARIABLE MAP--(LO=A)
 -NAME---ADDRESS --BLOCK------TYPE----          -NAME---ADDRESS --BLOCK-----PROPERTIES-------TYPE---
  A          220B             CHAR*1              IFXE        21B  /INTRS/                    INTEGER
  CR           0B  /CHARS/    CHAR*28             IFXS        20B  /INTRS/                    INTEGER
  FC         222B             CHAR*1              IFYE        17B  /INTRS/                    INTEGER
  I          231B             INTEGER             IFYS        16B  /INTRS/                    INTEGER
  IBLNK      223B             INTEGER             IMESH        4   DUMMY-ARG                  INTEGER
  ICHNG      224B             INTEGER             ISEAM        3   DUMMY-ARG                  INTEGER
  ICONVR       0B  /CHARS/    CHAR*1              ITAG       233B               *S*           INTEGER
  IFLAG        7B  /INTRS/    INTEGER             IWHICH      13B  /INTRS/                    INTEGER
  IFOUND      10B  /INTRS/    INTEGER             KODE         3B  /CHARS/                    CHAR*28
```

```
-NAME---ADDRESS --BLOCK------TYPE-----          -NAME---ADDRESS --BLOCK-----PROPERTIES-------TYPE----
 MAT           1  DUMMY-ARG  CHAR*(*)            NYEL        15B  /INTRS/                     INTEGER
 NCNTRL       4B  /INTRS/    INTEGER             NYMAX       12B  /INTRS/                     INTEGER
 NMATS        5B  /INTRS/    INTEGER             Q            0B  /CHARS/                     CHAR*1
 NMATS2       6B  /INTRS/    INTEGER             R          232B                              REAL
 NOSP         3B  /INTRS/    INTEGER             SKODE         2  DUMMY-ARG                   CHAR*(*)
 NSEAM        0B  /INTRS/    INTEGER             VR         221B                              CHAR*1
 NX           1B  /INTRS/    INTEGER             XMAX       226B                              REAL
 NXEL        14B  /INTRS/    INTEGER             XMIN       225B                              REAL
 NXMAX       11B  /INTRS/    INTEGER             YMAX       230B                              REAL
 NY           2B  /INTRS/    INTEGER             YMIN       227B                              REAL
--PROCEDURES--(LO=A)
 -NAME------TYPE--------ARGS------CLASS-----    -NAME------TYPE--------ARGS------CLASS-----
 ANMODE                    0      SUBROUTINE     LINEF                     0      SUBROUTINE
 CHANGE                   11      SUBROUTINE     PLTCRS                    4      SUBROUTINE
 HDCOPY                    0      SUBROUTINE     PRMAT                     8      SUBROUTINE
 HOME                      0      SUBROUTINE     RDTERM                    4      SUBROUTINE
 IOWAIT                    1      SUBROUTINE
--STATEMENT LABELS--(LO=A)
 -LABEL-ADDRESS-----PROPERTIES----DEF
   230      24B                    19
   250 *NO REFS*                   25
   255      61B                    31
   999 *NO REFS*                   40
--ENTRY POINTS--(LO=A)
 -NAME---ADDRESS--ARGS---
 MAKCRM       3B     4
--STATISTICS--
 PROGRAM-UNIT LENGTH               240B =   160
 CM LABELLED COMMON LENGTH          30B =   24
 CM STORAGE USED                 61000B =   25088
 COMPILE TIME                    0.322 SECONDS

       SUBROUTINE MAKEXT 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587  86/02/06. 14.17.54  PAGE 1
       DO=-LONG/-OT,ARG=-COMMON/-FIXED,CS= USER/-FIXED,DB=-TB/-SB/-SL/-ER/-ID/-PMD/-ST,PL=5000
       FTN5,I=PGNMLBM,B=BGNMLBM,L=LGNMLBM,OPT=3.
```

```
C***********************************************************
C     SUBROUTINE TO MAKE COARSE GRID EXTRACTION RATIOS
C***********************************************************
      SUBROUTINE MAKEXT(MAT, SKODE, ISEAM, IMESH)
      CHARACTER A, VR, MAT*(*), FC, SKODE*(*)
      CHARACTER Q, ICONVR, CR*28, KODE*28
      COMMON /CHARS/ Q, ICONVR, CR, KODE
      COMMON /INTRS/ NSEAM, NX, NY, NOSP, NCNTRL, NMATS, NMATS2,
     #               IFLAG, IFOUND, NXMAX, NYMAX, IWHICH, NXEL, NYEL
     #              ,IFYS,IFYE,IFXS,IFXE
      IBLNK = 0
      ICHNG = 1
        XMIN = 0
        XMAX = 60
        YMIN = 0
        YMAX = 60
        CALL PLTCRS(NX,NY,ISEAM,4)
        CALL PRMAT(MAT, ISEAM, XMIN, XMAX, YMIN, YMAX, IMESH, IBLNK)
  230   CALL HOME
        CALL LINEF
        CALL ANMODE
        PRINT *,'( $ = RIGID MATERIAL)'
        CALL ANMODE
        PRINT *,' CHANGES(Y/N)'
        CALL RDTERM(1,A,I,R)
        IF(A .NE. 'Y' .AND. A .NE. 'N') GO TO 230
  250   IF (A .EQ. 'Y') THEN
          CALL CHANGE(MAT, FC, VR, ISEAM,XMIN,XMAX,YMIN,YMAX,
     #               IMESH, ICHNG, SKODE)
          GO TO 230
        END IF
  255   PRINT *, ' DO YOU WANT'
        PRINT *, ' A HARD COPY(Y/N)'
        CALL RDTERM(1,A,I,R)
        IF(A .NE. 'Y' .AND. A .NE. 'N') GO TO 255
        IF(A .EQ. 'Y') THEN
          CALL IOWAIT(15)
```

```
                    CALL HDCOPY
                    CALL IOWAIT(25)
                  END IF
          999     RETURN
                  END
--VARIABLE MAP--(LO=A)
 -NAME---ADDRESS --BLOCK--------------TYPE----              -NAME---ADDRESS --BLOCK--------------TYPE----
  A           232B                   CHAR*1                  IFOUND       10B  /INTRS/             INTEGER
  CR            0B  /CHARS/          CHAR*28                 IFXE         21B  /INTRS/             INTEGER
  FC          234B                   CHAR*1                  IFXS         20B  /INTRS/             INTEGER
  I           243B                   INTEGER                 IFYE         17B  /INTRS/             INTEGER
  IBLNK       235B                   INTEGER                 IFYS         16B  /INTRS/             INTEGER
  ICHNG       236B                   INTEGER                 IMESH         4   DUMMY-ARG           INTEGER
  ICONVR        0B  /CHARS/          CHAR*1                  ISEAM         3   DUMMY-ARG           INTEGER
  IFLAG         7B  /INTRS/          INTEGER                 IWHICH       13B  /INTRS/             INTEGER

       SUBROUTINE MAKEXT 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587  86/02/06. 14.17.54  PAGE 2
 -NAME---ADDRESS --BLOCK--------------TYPE----              -NAME---ADDRESS --BLOCK--------------TYPE----
  KODE          3B  /CHARS/          CHAR*28                 NYEL         15B  /INTRS/             INTEGER
  MAT           1   DUMMY-ARG        CHAR*(*)                NYMAX        12B  /INTRS/             INTEGER
  NCNTRL        4B  /INTRS/          INTEGER                 Q             0B  /CHARS/             CHAR*1
  NMATS         5B  /INTRS/          INTEGER                 R           244B                      REAL
  NMATS2        6B  /INTRS/          INTEGER                 SKODE         2   DUMMY-ARG           CHAR*(*)
  NOSP          3B  /INTRS/          INTEGER                 VR          233B                      CHAR*1
  NSEAM         0B  /INTRS/          INTEGER                 XMAX        240B                      REAL
  NX            1B  /INTRS/          INTEGER                 XMIN        237B                      REAL
  NXEL         14B  /INTRS/          INTEGER                 YMAX        242B                      REAL
  NXMAX        11B  /INTRS/          INTEGER                 YMIN        241B                      REAL
  NY            2B  /INTRS/          INTEGER
--PROCEDURES--(LO=A)
 -NAME------TYPE--------ARGS------CLASS-----     -NAME------TYPE--------ARGS------CLASS-----
  ANMODE                   0      SUBROUTINE      LINEF                    0      SUBROUTINE
  CHANGE                  11      SUBROUTINE      PLTCRS                   4      SUBROUTINE
  HDCOPY                   0      SUBROUTINE      PRMAT                    8      SUBROUTINE
  HOME                     0      SUBROUTINE      RDTERM                   4      SUBROUTINE
  IOWAIT                   1      SUBROUTINE
```

```
--STATEMENT LABELS--(LO=A)
 -LABEL-ADDRESS-----PROPERTIES----DEF
    230     25B                     19
    250 *NO REFS*                   27
    255     65B                     32
    999 *NO REFS*                   41
--ENTRY POINTS--(LO=A)
 -NAME---ADDRESS--ARGS---
  MAKEXT      3B     4
--STATISTICS--
  PROGRAM-UNIT LENGTH             251B =  169
  CM LABELLED COMMON LENGTH        30B =  24
  CM STORAGE USED               61000B =  25088
  COMPILE TIME                  0.329 SECONDS
```

```
SUBROUTINE PLTCRS 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587  86/02/06. 14.17.54  PAGE 1
DO=-LONG/-OT,ARG=-COMMON/-FIXED,CS= USER/-FIXED,DB=-TB/-SB/-SL/-ER/-ID/-PMD/-ST,PL=5000
FTN5,I=PGNMLBM,B=BGNMLBM,L=LGNMLBM,OPT=3.
      SUBROUTINE PLTCRS(NX,NY,ISEAM,ITAG)
C
C..SUBROUTINE TO PLOT THE COARSE MESH GRID BLOCKS
C
      CALL NEWPAG
      CALL TWINDO(213,640,10,730)
      CALL DWINDO(0.,50.,0.,50.)
      RX = NX
      RY = NY
      DO 100 I = 0,RX
        RI = I
        CALL MOVEA(RI,0.)
        IF(I.GT.0.AND.I/5*5.EQ.I) THEN
          CALL MOVREL(-15,-15)
          ENCODE(2,1100,IARAY)I
          CALL AOUTST(2,IARAY)
          CALL MOVEA(RI,0.)
        END IF
```

```
          CALL DRAWA(RI,RY)
  100 CONTINUE
        DO 200 I = 0,RY
          RI = I
          CALL MOVEA(0.,RI)
          IF(I.GT.0.AND.I/5*5.EQ.I) THEN
            CALL MOVREL(-20,-10)
            ENCODE(2,1100,IARAY)I
            CALL AOUTST(2,IARAY)
            CALL MOVEA(0.,RI)
          END IF
          CALL DRAWA(RX,RI)
  200 CONTINUE
        DO 300 I=0,RX,5
          RI = I
          RI = RI - 0.125
          RI2 = RI + 0.125
          CALL MOVEA(RI,0.)
          CALL DRAWA(RI,RY)
          CALL MOVEA(RI2,0.)
          CALL DRAWA(RI2,RY)
  300 CONTINUE
        DO 400 I=0,RY,5
          RI = I
          RI = RI - 0.1
          RI2 = RI + 0.1
          CALL MOVEA(0.,RI)
          CALL DRAWA(RX,RI)
          CALL MOVEA(0.,RI2)
          CALL DRAWA(RX,RI2)
  400 CONTINUE
        CALL HOME
        CALL ANMODE
        CALL IOWAIT(15)
        IF(ITAG .EQ. 1) THEN
          PRINT 1200
```

```
 55                ELSE IF(ITAG.EQ.3) THEN

   SUBROUTINE PLTCRS 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587  86/02/06. 14.17.54  PAGE 2
 56                  PRINT 1000,ISEAM
 57                ELSE IF(ITAG.EQ.4) THEN
 58                        PRINT 1050,ISEAM
 59                END IF
 60           1000 FORMAT(T26,'COARSE MESH',/,T26,'MATERIAL PROPERTIES SEAM ',I3)
 61           1050 FORMAT(T26,'COARSE MESH',/,T26,'EXTRACTION RATIO CODES SEAM ',I3)
 62           1100 FORMAT(I2)
 63           1200 FORMAT(T26,'GRID MODEL (* = FINE MESH AREA)')
 64                CALL MOVEA(0.,0.)
 65                RETURN
 66                END
--VARIABLE MAP--(LO=A)
 -NAME---ADDRESS --BLOCK--------------TYPE---            -NAME---ADDRESS --BLOCK--------------TYPE---
  I          347B                      INTEGER              NY           2   DUMMY-ARG           INTEGER
  IARAY      352B                      INTEGER              RI         351B                      REAL
  ISEAM         3   DUMMY-ARG          INTEGER              RI2        355B                      REAL
  ITAG          4   DUMMY-ARG          INTEGER              RX         345B                      REAL
  NX            1   DUMMY-ARG          INTEGER              RY         346B                      REAL
--PROCEDURES--(LO=A)
 -NAME------TYPE--------ARGS------CLASS-----      -NAME------TYPE--------ARGS------CLASS-----
  ANMODE                   0      SUBROUTINE       IOWAIT                   1      SUBROUTINE
  AOUTST                   2      SUBROUTINE       MOVEA                    2      SUBROUTINE
  DRAWA                    2      SUBROUTINE       MOVREL                   2      SUBROUTINE
  DWINDO                   4      SUBROUTINE       NEWPAG                   0      SUBROUTINE
  HOME                     0      SUBROUTINE       TWINDO                   4      SUBROUTINE
--STATEMENT LABELS--(LO=A)
 -LABEL-ADDRESS-----PROPERTIES----DEF       -LABEL-ADDRESS-----PROPERTIES----DEF
   100  INACTIVE   DO-TERM        20         1000     220B     FORMAT          60
   200  INACTIVE   DO-TERM        31         1050     227B     FORMAT          61
   300  INACTIVE   DO-TERM        40         1100     236B     FORMAT          62
   400  INACTIVE   DO-TERM        49         1200     240B     FORMAT          63
--ENTRY POINTS--(LO=A)
```

```
-NAME---ADDRESS--ARGS---
 PLTCRS      3B     4
--STATISTICS--
 PROGRAM-UNIT LENGTH             357B =  239
 CM STORAGE USED               61000B =  25088
 COMPILE TIME                  0.490 SECONDS
```

SUBROUTINE WRTMAT 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587 86/02/06. 14.17.54 PAGE 1

```
DO=-LONG/-OT,ARG=-COMMON/-FIXED,CS= USER/-FIXED,DB=-TB/-SB/-SL/-ER/-ID/-PMD/-ST,PL=5000
FTN5,I=PGNMLBM,B=BGNMLBM,L=LGNMLBM,OPT=3.
C*******************************************************************
C     SUBROUTINE WRTMAT
C*******************************************************************
      SUBROUTINE WRTMAT(MAT,IMESH)
C
C.. IMESH = 0 (FINE MESH)
C           1 (COARSE MESH)
      CHARACTER *(*) MAT, IMAT*100
      COMMON /INTRS/ NSEAM, NX, NY, NOSP, NCNTRL, NMATS, NMATS2,
     #               IFLAG, IFOUND, NXMAX, NYMAX, IWHICH, NXEL, NYEL
     #               ,IFYS,IFYE,IFXS,IFXE
C
C *** BUILD ARRAY WITH MATERIAL/STATUS CODES IN CORRECT
C     ORDER FOR DISPLAYING
C
      NXSTRT = 1
      NYEND  = 1
      IF(IMESH .EQ. 0) THEN
        NYSTRT = NYEL
        NXEND  = NXEL
      ELSE
        NYSTRT = NY
        NXEND  = NX
      END IF
      DO 2000 K = NYSTRT,NYEND,-1
        LCNT = NYSTRT - K
```

```
                 L = 0
                 IMAT = ' '
                  DO 1000 I = NXSTRT,NXEND
                    J = I * NYSTRT - LCNT
                    L = L + 1
                    IMAT(L:L) = MAT(J:J)
          1000 CONTINUE
         C
         C ***  WRITE MATERIAL/STATUS/EXTRACTION CODES TO THE DISK FILE
         C
                    WRITE(2,3000)IMAT(1:L)
          3000 FORMAT(A)
          2000 CONTINUE
               RETURN
               END
--VARIABLE MAP--(LO=A)
 -NAME---ADDRESS --BLOCK--------------TYPE----          -NAME---ADDRESS --BLOCK--------------TYPE---
  I          144B                     INTEGER            IWHICH       13B  /INTRS/             INTEGER
  IFLAG        7B  /INTRS/            INTEGER            J           145B                      INTEGER
  IFOUND      10B  /INTRS/            INTEGER            K           140B                      INTEGER
  IFXE        21B  /INTRS/            INTEGER            L           143B                      INTEGER
  IFXS        20B  /INTRS/            INTEGER            LCNT        142B                      INTEGER
  IFYE        17B  /INTRS/            INTEGER            MAT           1   DUMMY-ARG           CHAR*(*
  IFYS        16B  /INTRS/            INTEGER            NCNTRL       4B   /INTRS/             INTEGER
  IMAT       122B                     CHAR*100           NMATS        5B   /INTRS/             INTEGER
  IMESH        2   DUMMY-ARG          INTEGER            NMATS2       6B   /INTRS/             INTEGER

       SUBROUTINE WRTMAT 74/855 OPT=3,ROUND= A/ S/ M/-D,-DS FTN 5.1+587  86/02/06. 14.17.54  PAGE 2
 -NAME---ADDRESS --BLOCK--------------TYPE----          -NAME---ADDRESS --BLOCK--------------TYPE---
  NOSP         3B  /INTRS/            INTEGER            NXSTRT      134B                      INTEGER
  NSEAM        0B  /INTRS/            INTEGER            NY           2B   /INTRS/             INTEGER
  NX           1B  /INTRS/            INTEGER            NYEL        15B   /INTRS/             INTEGER
  NXEL        14B  /INTRS/            INTEGER            NYEND       135B                      INTEGER
  NXEND      137B                     INTEGER            NYMAX        12B  /INTRS/             INTEGER
  NXMAX       11B  /INTRS/            INTEGER            NYSTRT      136B                      INTEGER
--STATEMENT LABELS--(LO=A)
```

```
-LABEL-ADDRESS-----PROPERTIES----DEF
  1000  INACTIVE   DO-TERM        33
  2000  INACTIVE   DO-TERM        39
  3000     74B     FORMAT         38
--ENTRY POINTS--(LO=A)
 -NAME---ADDRESS--ARGS---
 WRTMAT      3B     2
--I/O UNITS--(LO=A)
 -NAME--- PROPERTIES-------------
 TAPE2   FMT/SEQ
--STATISTICS--
 PROGRAM-UNIT LENGTH             146B =  102
 CM LABELLED COMMON LENGTH        22B =  18
 CM STORAGE USED               61000B =  25088
 COMPILE TIME                  0.264 SECONDS
```

APPENDIX E.--SAMPLE PROBLEM INPUT

This is the file generated by the sample session of appendix B.

```
MULSIM/BM SAMPLE PROBLEM -- THIS IS LINE ONE OF THE FILE
 .150000 .400000E+06       2
  2
  .2000000000E+06 .7692300000E+05 .00E+00 .00E+00 .00E+00 .00E+00       0
  .5000000000E+04 .1786000000E+04 .00E+00 .00E+00 .00E+00 .00E+00       1
    0. .0458    0. .0000    0. .0000    0. .0458    0. .0000    0. .0917
  1200.00       8       8       1       1       4       5       2       7
       1    .200    .000    .000    .000    .000    .000    .000    .000    .000
      .0      .0-12960.0   120.0
      .0      .0-12000.0    96.0
  1.00000  .00000  .00000  .00000 1.00000  .00000  .00000  .00000 1.00000
     1.35     100       1       0       0
AAAAAAAAAA
AAAAAAAAAA
AAAAAAAAAA
AAAAAAAAAA
1111111111
AAAAAAAAAA
AAAAAAAAAA
AAAAAAAAAA
AAAAAAAAAA
AAAAAAAAAA
AAAAAAAAAA
AAAAAAAAAA
AAAAAAAAAA
AAAAAAAAAA
AAAAAAAAAA
AAAAAAAAAA
AAAAAAAAAA
AAAAAAAAAA
AAAAAAAAAA
AAAAAAAAAA
AAAAAAAAAA
AAAAAAAAAA
AAAAAAAAAA
AAAAAAAAAA
AAAAAAAAAA
1111111111
AAAAAAAAAA
AAAAAAAAAA
AAAAAAAAAA
AAAAAAAAAA
AAAAAAAAAA
AAAAAAAAAA
```

```
AAAAAAAAAA
AAAAAAAAAA
1111111111
AAAA1BBBBB
AAAA1BBBBB
AAAA1BBBBB
AAAA1BBBBB
AAAA1BBBBB
AAAA1BBBBB
AAAA1BBBBB
AAAA1BBBBB
AAAA1BBBBB
AAAA1BBBBB
AAAA1BBBBB
AAAA1BBBBB
AAAA1BBBBB
AAAA1BBBBB
AAAA1BBBBB
AAAA1BBBBB
AAAA1BBBBB
AAAA1BBBBB
AAAA1BBBBB
AAAA1BBBBB
1111111111
AAAAAAAAAA
AAAAAAAAAA
AAAAAAAAAA
AAAAAAAAAA
3333333333
3333333333
3333333333
3333333333
1111111111
3333333333
3333333333
3333333333
3333333333
3333333333
3333333333
3333333333
3333333333
3333333333
3333333333
3333333333
3333333333
3333333333
3333333333
3333333333
3333333333
```

```
3333333333
3333333333
3333333333
3333333333
1111111111
3333333333
3333333333
3333333333
3333333333
3333333333
3333333333
3333333333
3333333333
1111111111
3333133333
3333133333
3333133333
3333133333
3333133333
3333133333
3333133333
3333133333
3333133333
3333133333
3333133333
3333133333
3333133333
3333133333
3333133333
3333133333
3333133333
3333133333
3333133333
3333133333
1111111111
3333333333
3333333333
3333333333
3333333333
AAAAAAAA
AAA**AAA
AAA**AAA
AAA**AAA
AAA**AAA
AAA**AAA
AAA**AAA
AAAAAAAA
AAAAAAAA
AAA**AAA
```

```
AAA**BBB
AAA**BBB
AAA**BBB
AAA**BBB
AAA**AAA
AAAAAAAA
00000000
111**111
000**000
000**000
000**000
000**000
111**111
00000000
00000000
111**111
000**000
000**000
000**000
000**000
111**111
00000000
 .360000E+04 .480000E+04-.130920E+05 .240000E+03 .240000E+03      10      01
 1.00000  .00000  .00000  .00000 1.00000  .00000  .00000  .00000 1.00000
```

APPENDIX F.--SAMPLE PROBLEM OUTPUT

```
$$      $$  $$$$$$$$$$  $$      $$  $$$$$$$$$$   $$$$$$$$               $$$$$$$$$   $$$$$$$$$    $$$$$$$$
$$$$  $$$$  $$$$$$$$$$  $$$     $$  $$$$$$$$$$  $$$$$$$$$$              $$$$$$$$$$  $$$$$$$$$$  $$$$$$$$$$
$$ $$$$ $$      $$      $$ $    $$  $$          $$       $              $$      $$  $$      $$  $$       $
$$  $$  $$      $$      $$ $$   $$  $$          $$                      $$      $$  $$      $$  $$
$$  $$  $$      $$      $$  $$  $$  $$$$$       $$$$$$$$$   $$$$$$$$$$  $$      $$  $$$$$$$$$$  $$
$$      $$      $$      $$  $$  $$  $$$$$        $$$$$$$$$  $$$$$$$$$$  $$      $$  $$$$$$$$$   $$
$$      $$      $$      $$   $$ $$  $$                  $$              $$      $$  $$   $$     $$
$$      $$      $$      $$    $ $$  $$          $       $$              $$      $$  $$    $$    $$       $
$$      $$  $$$$$$$$$$  $$     $$$  $$$$$$$$$$  $$$$$$$$$$              $$$$$$$$$$  $$     $$   $$$$$$$$$$
$$      $$  $$$$$$$$$$  $$      $$  $$$$$$$$$$   $$$$$$$$               $$$$$$$$$   $$      $$   $$$$$$$$

$$$$$$$$$   $$      $$   $$$$$$$$   $$$$$$$$$   $$$$$$$$$   $$$$$$$$$$  $$$$$$$$$
$$$$$$$$$$  $$$$  $$$$  $$$$$$$$$$  $$$$$$$$$$  $$$$$$$$$$  $$$$$$$$$$  $$$$$$$$$$
$$      $$  $$ $$$$ $$  $$      $$  $$      $$  $$      $$      $$      $$      $$
$$      $$  $$  $$  $$  $$      $$  $$      $$  $$      $$      $$      $$      $$
$$$$$$$$$$  $$  $$  $$  $$      $$  $$      $$  $$$$$$$$$$      $$      $$      $$
$$$$$$$$$   $$      $$  $$$$$$$$$$  $$      $$  $$$$$$$$$       $$      $$      $$
$$   $$     $$      $$  $$$$$$$$$$  $$      $$  $$   $$         $$      $$      $$
$$    $$    $$      $$  $$      $$  $$      $$  $$    $$        $$      $$      $$
$$     $$   $$      $$  $$      $$  $$$$$$$$$$  $$     $$   $$$$$$$$$$  $$$$$$$$$$
$$      $$  $$      $$  $$      $$  $$$$$$$$$   $$      $$  $$$$$$$$$$  $$$$$$$$$

 $$$$$$$$   $$$$$$$$$$   $$$$$$$$      $$$$     $$$$$$$$$$   $$$$$$$$       $$$
$$$$$$$$$$  $$$$$$$$$   $$$$$$$$$$    $$$$$$    $$$$$$$$$$  $$$$$$$$$$     $$$$
$       $$        $$    $$       $   $$    $$          $$   $$       $    $$ $$
        $$       $$     $$          $$      $$        $$    $$           $$  $$
        $$      $$$     $$$$$$$$$   $$      $$       $$     $$$$$$$$$   $$   $$
      $$          $$$   $$$$$$$$$$  $$      $$      $$      $$$$$$$$$$  $$$$$$$$$$
    $$             $$   $$      $$  $$      $$     $$       $$      $$  $$$$$$$$$$
   $$       $      $$   $$      $$   $$    $$     $$        $$      $$       $$
$$$$$$$$$$  $$$$$$$$$$  $$$$$$$$$$    $$$$$$    $$          $$$$$$$$$$       $$
$$$$$$$$$$   $$$$$$$$    $$$$$$$$      $$$$     $$           $$$$$$$$        $$
```

```
MULSIM/BM SAMPLE PROBLEM -- THIS IS LINE ONE OF THE FILE

  IN THE FOLLOWING X,Y,Z REFER TO GLOBAL AXES OR COORDINATES
  AND X1,X2,X3  OR  1,2,3 REFER TO THE LOCAL AXES OR COORDINATES.

  MATERIAL PROPERTIES   --

  POISSON"S RATIO OF ROCK MASS          --            .15

  MODULUS OF ELASTICITY OF ROCK MASS    --      400000.00

  NO. OF SEAMS                               2

  NO. OF MATERIALS                           2

 MATERIAL PROPERTY SET NUMBER --  1

  MODULUS OF ELASTICITY OF SEAM MATERIAL--      200000.00

  MODULUS OF RIGIDITY OF SEAM MATERIAL  --       76923.00

  COHESION OF THE SEAM MATERIAL         --           0.00

  FRICTION ANGLE OF THE SEAM MATERIAL   --           0.00

  RESIDUAL VALUE OF THE COHESION        --           0.00

  RESIDUAL VALUE OF THE FRICTION ANGLE  --           0.00

 MATERIAL PROPERTY SET NUMBER --  2

   THIS IS A GOB OR INSERTED MATERIAL --

  MODULUS OF ELASTICITY OF SEAM MATERIAL--        5000.00

  MODULUS OF RIGIDITY OF SEAM MATERIAL  --        1786.00

  COHESION OF THE SEAM MATERIAL         --           0.00

  FRICTION ANGLE OF THE SEAM MATERIAL   --           0.00

  RESIDUAL VALUE OF THE COHESION        --           0.00

  RESIDUAL VALUE OF THE FRICTION ANGLE  --           0.00

  PRIMITIVE STRESS PARAMETERS  --

  PRIMITIVE STRESSES ARE GIVEN AS--
  PXX=  0.0000--   .0458 Z
  PXY=  0.0000--  0.0000 Z
  PXZ=  0.0000--  0.0000 Z
  PYY=  0.0000--   .0458 Z
  PYZ=  0.0000--  0.0000 Z
  PZZ=  0.0000--   .0917 Z

  MODEL CONTROL AND MODEL DATA  --

  NO. OF PARALLEL SEAMS              --
```

```
WIDTH OF BLOCKS                      -- 1200.00

NO. OF BLOCKS ALONG X1 AXIS          --       8

NO. OF BLOCKS ALONG X2 AXIS          --       8

SYMMETRY CODE SPECIFIED              --       1

NUMBER OF OFF-SEAM PLANES SELECTED   --       1

FINE MESH STARTING BLOCK X-AXIS      --       4

FINE MESH ENDING   BLOCK X-AXIS      --       5

FINE MESH STARTING BLOCK Y-AXIS      --       2

FINE MESH ENDING   BLOCK Y-AXIS      --       7


NUMBER OF EXTRACTION CODES USED IN MODEL --       1

EXTRACTION RATIO CODE TABLE :       .200

GLOBAL COORDINATES OF THE LOCAL ORIGINS AND SEAM THICKNESSES --
SEAM NO.         X           Y           Z      THICKNESS
     1        0.00        0.00   -12960.00       120.00
     2        0.00        0.00   -12000.00        96.00

ORIENTATION OF THE SEAMS- DIRECTION COSINES OF THE LOCAL AXES
WITH RESPECT TO THE GLOBAL AXES --

   EN(1,X)    EN(2,X)    EN(3,X)    EN(1,Y)    EN(2,Y)    EN(3,Y)    EN(1,Z)    EN(2,Z)    EN(3,Z)
   1.00000    0.00000    0.00000    0.00000    1.00000    0.00000    0.00000    0.00000    1.00000

PROGRAM FLOW CONTROL PARAMETERS  --

OVER RELAXATION FACTOR               --    1.35

MAXIMUM NO. OF ITERATIONS SPECIFIED  --      20

THIS IS THE MAIDEN COMPUTER RUN FOR THIS PROBLEM

THIS IS A NEW GRID, INFLUENCE COEFFICIENTS ARE COMPUTED WITHIN THIS COMPUTER RUN
```

```
MINE SEAM PROPERTIES:
    MINED OUT = 1
        RIGID = 2
SEAM PROPERTIES ARE:
A B
 NO. OF SEAM-    1

        AAAAA AAAAA
        AAAAA AAAAA
        AAAAA AAAAA
        AAAAA AAAAA
        11111 11111

        AAAAA AAAAA
        AAAAA AAAAA
        AAAAA AAAAA
        AAAAA AAAAA
        AAAAA AAAAA

        AAAAA AAAAA
        AAAAA AAAAA
        AAAAA AAAAA
        AAAAA AAAAA
        AAAAA AAAAA

        AAAAA AAAAA
        AAAAA AAAAA
        AAAAA AAAAA
        AAAAA AAAAA
        AAAAA AAAAA

        AAAAA AAAAA
        AAAAA AAAAA
        AAAAA AAAAA
        AAAAA AAAAA
        AAAAA AAAAA

        11111 11111
        AAAAA AAAAA
        AAAAA AAAAA
        AAAAA AAAAA
        AAAAA AAAAA

 NO. OF SEAM-    2

        AAAAA AAAAA
        AAAAA AAAAA
        AAAAA AAAAA
        AAAAA AAAAA
        11111 11111

        AAAA1 BBBBB
        AAAA1 BBBBB
        AAAA1 BBBBB
        AAAA1 BBBBB
        AAAA1 BBBBB
```

```
          AAAA1 BBBBB
          AAAA1 BBBBB
          AAAA1 BBBBB
          AAAA1 BBBBB
          AAAA1 BBBBB

          AAAA1 BBBBB
          AAAA1 BBBBB
          AAAA1 BBBBB
          AAAA1 BBBBB
          AAAA1 BBBBB

          AAAA1 BBBBB
          AAAA1 BBBBB
          AAAA1 BBBBB
          AAAA1 BBBBB
          AAAA1 BBBBB

          11111 11111
          AAAAA AAAAA
          AAAAA AAAAA
          AAAAA AAAAA
          AAAAA AAAAA

MINING KODES      -   MINED OUT (1)
                      UNMINED, SEAM MATERIAL -
                      RIGID (2)
                      ELASTIC (3)
                      MOHR COULOMB TYPE (4)
                      MOHR COULOMB TYPE-YIELDED (5)
                      MOHR COULOMB TYPE-CRACKED (6)
                      MOHR COULOMB TYPE-CRACKED AND   CLOSED BACK (7)
                      MOHR COULOMB TYPE-CRACKED OPEN   CLOSED BACK AND YIELDED (8)

NO. OF SEAM-   1

          33333 33333
          33333 33333
          33333 33333
          33333 33333
          11111 11111

          33333 33333
          33333 33333
          33333 33333
          33333 33333
          33333 33333

          33333 33333
          33333 33333
          33333 33333
          33333 33333
          33333 33333

          33333 33333
          33333 33333
          33333 33333
          33333 33333
          33333 33333
```

```
        33333 33333
        33333 33333
        33333 33333
        33333 33333
        33333 33333

        11111 11111
        33333 33333
        33333 33333
        33333 33333
        33333 33333

NO. OF SEAM-    2

        33333 33333
        33333 33333
        33333 33333
        33333 33333
        11111 11111

        33331 33333
        33331 33333
        33331 33333
        33331 33333
        33331 33333

        33331 33333
        33331 33333
        33331 33333
        33331 33333
        33331 33333

        33331 33333
        33331 33333
        33331 33333
        33331 33333
        33331 33333

        33331 33333
        33331 33333
        33331 33333
        33331 33333
        33331 33333

        11111 11111
        33333 33333
        33333 33333
        33333 33333
        33333 33333
```

```
MINE SEAM PROPERTIES: FOR COARSE MESH

    MINED OUT = 1

        RIGID = 2

SEAM PROPERTIES ARE:

A B



 NO. OF SEAM-    1

         AAAAAAAA
         AAA**AAA
         AAA**AAA
         AAA**AAA
         AAA**AAA
         AAA**AAA
         AAA**AAA
         AAAAAAAA

 NO. OF SEAM-    2

         AAAAAAAA
         AAA**AAA
         AAA**BBB
         AAA**BBB
         AAA**BBB
         AAA**BBB
         AAA**AAA
         AAAAAAAA

    EXTRACTION RATIO CODES FOR COARSE MESH

 NUMBER OF EXTRACTION CODES USED IN MODEL --       1

 EXTRACTION RATIO CODE TABLE :       .200



 NO. OF SEAM-    1

         00000000
         111**111
         000**000
         000**000
         000**000
         000**000
         111**111
         00000000

 NO. OF SEAM-    2

         00000000
         111**111
```

```
        000**000
        000**000
        000**000
        000**000
        111**111
        00000000

TIME TAKEN TO READ AND PRINT INPUT VARIABLES                     --    .5330 SECOND

TIME TAKEN TO DEFINE THE CONSTANTS, INITIALIZE THE CLOSURE AND
RIDE VALUES AND OBTAIN THE PRIMITIVE STRESSES                    --    .0210 SECOND

TIME TAKEN TO COMPUTE/RETRIEVE THE INFLUENCE COEFFICIENTS        --  ' 1.6400 SECOND

TIME TAKEN TO SOLVE FOR UNKNOWN CLOSURES AND RIDES               -- 185.5220 SECOND

NO. OF ITERATIONS COMPLETED IN ALL THE RUNS   =     12

MAXIMUM ERROR IN THE RIDE AND CLOSURE VALUES =  .0007998

CLOSURE, RIDE AND DISPLACEMENT VALUES ARE IN THE UNITS  OF INPUT LINEAR DIMENSIONS
(VIZ,XO,YO,ZO,HW...ETC) AND STRESSES ARE IN THE UNITS OF MODULUS OF ELASTICITY (E).

MAXIMUM ERROR IN THE RIDE AND CLOSURE BLOCK VALUES =  .0000805
```

NO.OF SEAM-- 1

BLOCK		ELEM														
COL	ROW	COL	ROW	CODE	RIDE1	RIDE2	CLOSURE	U1POS	U2POS	U3POS	U1NEG	U2NEG	U3NEG	SIGS1	SIGS2	SIGN
4	2	1	1	3	.017	.007	.097	-.303	-.325	.064	-.286	-.318	.161	10.80	4.49	1350.77
4	2	1	2	3	.009	-.002	.109	-.295	-.314	.089	-.287	-.315	.197	5.57	-1.04	1369.86
4	2	1	3	3	-.007	-.011	.126	-.290	-.316	.117	-.297	-.327	.243	-4.48	-6.87	1398.60
4	2	1	4	3	-.032	-.012	.225	-.300	-.373	.106	-.332	-.386	.330	-20.43	-8.00	1562.60
4	2	1	5	1	-.103	-.001	2.214	-.313	-.261	-.863	-.416	-.263	1.351	-.00	-.00	-.00
4	2	2	1	3	.014	-.001	.095	-.298	-.316	.081	-.284	-.317	.176	8.78	-.66	1346.36
4	2	2	2	3	.001	-.017	.106	-.285	-.301	.115	-.283	-.319	.221	.76	-11.02	1365.28
4	2	2	3	3	-.023	-.035	.127	-.263	-.304	.154	-.286	-.339	.281	-14.63	-22.59	1400.87
4	2	2	4	3	-.064	-.045	.252	-.234	-.372	.145	-.298	-.417	.397	-40.93	-28.86	1607.97
4	2	2	5	1	-.197	-.059	2.445	-.154	-.217	-.915	-.351	-.276	1.530	-.00	-.00	-.01
4	2	3	1	3	.010	-.012	.097	-.287	-.304	.096	-.278	-.315	.194	6.13	-7.38	1350.56
4	2	3	2	3	-.007	-.038	.108	-.267	-.279	.140	-.274	-.317	.248	-4.69	-24.12	1368.75
4	2	3	3	3	-.040	-.069	.128	-.231	-.269	.193	-.270	-.338	.321	-25.49	-44.09	1401.10
4	2	3	4	3	-.099	-.094	.248	-.171	-.324	.204	-.269	-.418	.452	-63.20	-60.47	1602.06
4	2	3	5	1	-.304	-.159	2.419	-.028	-.135	-.834	-.332	-.294	1.585	.00	.00	.00
4	2	4	1	3	.006	-.024	.100	-.277	-.288	.111	-.271	-.312	.211	4.05	-15.46	1354.89
4	2	4	2	3	-.014	-.063	.109	-.252	-.248	.164	-.266	-.311	.273	-8.95	-40.23	1370.43
4	2	4	3	3	-.053	-.112	.122	-.204	-.217	.234	-.258	-.329	.356	-34.26	-71.78	1392.35
4	2	4	4	3	-.128	-.162	.226	-.124	-.245	.269	-.252	-.407	.495	-81.98	-104.03	1564.91
4	2	4	5	1	-.397	-.306	2.287	.067	-.015	-.708	-.330	-.321	1.579	-.01	.00	.00
4	2	5	1	3	.005	-.037	.103	-.271	-.270	.123	-.266	-.307	.226	3.43	-23.94	1360.55
4	2	5	2	3	-.016	-.090	.110	-.244	-.213	.185	-.260	-.303	.295	-10.23	-57.64	1372.30
4	2	5	3	3	-.058	-.161	.115	-.194	-.156	.268	-.252	-.317	.383	-36.96	-103.02	1379.93
4	2	5	4	3	-.138	-.242	.196	-.109	-.151	.324	-.247	-.393	.520	-88.27	-155.32	1514.97
4	2	5	5	1	-.430	-.486	2.117	.092	.127	-.589	-.337	-.359	1.528	-.00	-.01	-.01
4	3	1	1	3	-.057	.015	.251	-.355	-.115	.136	-.411	-.100	.387	-36.23	9.71	1605.94
4	3	1	2	3	-.061	.021	.157	-.309	-.182	.180	-.369	-.161	.337	-38.80	13.55	1449.30
4	3	1	3	3	-.067	.020	.145	-.284	-.197	.181	-.351	-.177	.326	-42.80	12.93	1430.90
4	3	1	4	3	-.072	.018	.144	-.269	-.199	.178	-.341	-.181	.322	-46.37	11.43	1428.26
4	3	1	5	3	-.076	.016	.144	-.259	-.198	.175	-.335	-.182	.319	-48.76	10.25	1428.49
4	3	2	1	3	-.117	-.010	.282	-.238	-.054	.193	-.355	-.064	.475	-75.26	-6.16	1658.07
4	3	2	2	3	-.131	.005	.160	-.195	-.144	.256	-.326	-.140	.416	-84.19	3.15	1454.32
4	3	2	3	3	-.148	.009	.142	-.160	-.173	.263	-.308	-.165	.405	-95.16	5.45	1425.52
4	3	2	4	3	-.162	.009	.139	-.135	-.184	.263	-.297	-.174	.401	-103.86	6.08	1419.62
4	3	2	5	3	-.171	.011	.138	-.119	-.190	.261	-.290	-.179	.399	-109.48	6.76	1418.47
4	3	3	1	3	-.193	-.059	.261	-.117	.010	.281	-.310	-.049	.541	-123.78	-37.95	1622.86
4	3	3	2	3	-.223	-.031	.127	-.060	-.093	.358	-.283	-.124	.485	-143.10	-19.82	1400.28
4	3	3	3	3	-.256	-.017	.103	-.009	-.135	.372	-.265	-.153	.475	-163.86	-10.99	1359.98
4	3	3	4	3	-.280	-.008	.095	.028	-.158	.377	-.252	-.167	.472	-179.41	-5.17	1347.53
4	3	3	5	3	-.295	-.001	.092	.051	-.176	.379	-.245	-.176	.471	-189.37	-.36	1342.55
4	3	4	1	3	-.262	-.136	.200	-.020	.095	.375	-.283	-.041	.574	-168.00	-87.46	1521.19
4	3	4	2	3	-.309	-.088	.053	.053	-.021	.465	-.255	-.109	.517	-197.86	-56.51	1275.99
4	3	4	3	3	-.356	-.058	.017	.120	-.083	.487	-.236	-.140	.504	-228.05	-37.06	1216.22
4	3	4	4	3	-.390	-.035	.002	.167	-.123	.497	-.223	-.158	.499	-250.12	-22.69	1191.55
4	3	4	5	3	-.412	-.018	-.006	.198	-.156	.502	-.215	-.173	.496	-264.23	-11.32	1179.16
4	3	5	1	3	-.287	-.234	.125	.002	.195	.435	-.285	-.039	.560	-183.69	-149.92	1396.76
4	3	5	2	3	-.340	-.162	-.040	.080	.064	.528	-.260	-.098	.488	-218.01	-103.96	1121.95
4	3	5	3	3	-.394	-.111	-.092	.151	-.019	.552	-.243	-.130	.460	-252.56	-71.24	1035.14
4	3	5	4	3	-.434	-.072	-.117	.203	-.080	.562	-.231	-.152	.446	-277.88	-46.10	993.79
4	3	5	5	3	-.459	-.041	-.130	.235	-.131	.568	-.224	-.172	.438	-294.11	-26.32	971.57
4	4	1	1	3	-.073	.009	.163	-.309	-.106	.166	-.382	-.097	.329	-46.56	5.77	1460.59
4	4	1	2	3	-.075	.008	.167	-.306	-.095	.171	-.381	-.087	.337	-47.97	5.22	1466.34
4	4	1	3	3	-.076	.007	.168	-.304	-.084	.174	-.380	-.076	.342	-48.99	4.80	1468.60
4	4	1	4	3	-.077	.007	.168	-.303	-.072	.175	-.380	-.066	.343	-49.52	4.20	1468.39
4	4	1	5	3	-.077	.005	.167	-.303	-.062	.174	-.380	-.057	.340	-49.54	3.13	1466.00

NO.OF SEAM-- 1

BLOCK COL	BLOCK ROW	ELEM COL	ELEM ROW	CODE	RIDE1	RIDE2	CLOSURE	U1POS	U2POS	U3POS	U1NEG	U2NEG	U3NEG	SIGS1	SIGS2	SIGN
4	4	2	1	3	-.171	.005	.159	-.162	-.100	.252	-.333	-.095	.412	-109.63	3.08	1453.65
4	4	2	2	3	-.175	.006	.163	-.155	-.091	.259	-.330	-.086	.422	-112.49	3.53	1460.26
4	4	2	3	3	-.179	.006	.165	-.151	-.082	.264	-.329	-.076	.428	-114.43	3.84	1462.81
4	4	2	4	3	-.180	.006	.164	-.149	-.072	.265	-.329	-.066	.430	-115.40	3.82	1462.30
4	4	2	5	3	-.180	.005	.162	-.149	-.063	.264	-.329	-.058	.427	-115.43	3.28	1459.13
4	4	3	1	3	-.301	-.003	.114	.017	-.088	.370	-.284	-.092	.485	-192.92	-2.22	1379.21
4	4	3	2	3	-.308	.000	.118	.028	-.084	.380	-.280	-.084	.498	-197.70	.14	1385.45
4	4	3	3	3	-.313	.003	.119	.035	-.078	.386	-.278	-.075	.505	-200.76	1.85	1387.55
4	4	3	4	3	-.316	.005	.119	.038	-.071	.388	-.277	-.066	.507	-202.28	2.97	1386.39
4	4	3	5	3	-.316	.006	.116	.037	-.066	.387	-.279	-.060	.504	-202.37	3.53	1382.31
4	4	4	1	3	-.423	-.016	.016	.172	-.071	.494	-.251	-.087	.510	-271.08	-10.15	1214.99
4	4	4	2	3	-.433	-.008	.018	.187	-.074	.506	-.246	-.081	.524	-277.70	-4.91	1219.19
4	4	4	3	3	-.440	-.002	.019	.196	-.072	.513	-.244	-.074	.532	-281.86	-1.13	1219.86
4	4	4	4	3	-.443	.003	.017	.200	-.070	.517	-.243	-.067	.534	-283.94	1.67	1217.53
4	4	4	5	3	-.443	.006	.014	.199	-.069	.516	-.245	-.063	.531	-284.13	3.86	1212.37
4	4	5	1	3	-.472	-.032	-.111	.213	-.051	.558	-.259	-.082	.447	-302.77	-20.36	1003.84
4	4	5	2	3	-.484	-.018	-.111	.231	-.061	.572	-.254	-.079	.461	-310.49	-11.41	1003.77
4	4	5	3	3	-.492	-.008	-.112	.241	-.065	.581	-.251	-.073	.469	-315.30	-5.17	1001.85
4	4	5	4	3	-.496	-.001	-.114	.245	-.068	.585	-.250	-.068	.471	-317.75	-.40	998.22
4	4	5	5	3	-.496	.006	-.118	.244	-.073	.585	-.252	-.067	.467	-317.99	3.87	992.50
4	5	1	1	3	-.077	-.005	.167	-.303	.061	.174	-.380	.056	.341	-49.55	-3.08	1466.19
4	5	1	2	3	-.077	-.007	.168	-.303	.071	.175	-.380	.065	.343	-49.56	-4.17	1468.64
4	5	1	3	3	-.077	-.007	.168	-.304	.083	.174	-.380	.075	.343	-49.08	-4.80	1468.91
4	5	1	4	3	-.075	-.008	.167	-.306	.094	.171	-.381	.086	.338	-48.13	-5.28	1466.70
4	5	1	5	3	-.073	-.009	.164	-.309	.105	.167	-.382	.096	.330	-46.84	-5.91	1461.00
4	5	2	1	3	-.180	-.005	.163	-.149	.062	.265	-.329	.057	.427	-115.45	-3.26	1459.34
4	5	2	2	3	-.180	-.006	.164	-.149	.071	.266	-.329	.065	.430	-115.46	-3.83	1462.58
4	5	2	3	3	-.179	-.006	.165	-.150	.081	.264	-.329	.075	.429	-114.58	-3.91	1463.14
4	5	2	4	3	-.176	-.006	.163	-.154	.091	.260	-.330	.085	.424	-112.76	-3.71	1460.63
4	5	2	5	3	-.172	-.005	.159	-.161	.099	.254	-.333	.094	.413	-110.09	-3.40	1454.03
4	5	3	1	3	-.316	-.006	.116	.037	.065	.388	-.279	.059	.504	-202.40	-3.57	1382.53
4	5	3	2	3	-.316	-.005	.119	.038	.070	.389	-.277	.065	.508	-202.38	-3.06	1386.67
4	5	3	3	3	-.314	-.003	.120	.035	.077	.387	-.278	.074	.506	-200.97	-2.03	1387.86
4	5	3	4	3	-.309	-.001	.118	.029	.084	.381	-.280	.083	.499	-198.06	-.48	1385.74
4	5	3	5	3	-.302	.003	.115	.018	.089	.372	-.284	.091	.487	-193.55	1.62	1379.41
4	5	4	1	3	-.443	-.006	.014	.199	.069	.517	-.245	.062	.531	-284.16	-3.96	1212.59
4	5	4	2	3	-.443	-.003	.018	.200	.069	.517	-.243	.066	.535	-284.06	-1.84	1217.78
4	5	4	3	3	-.440	.001	.019	.196	.072	.514	-.244	.073	.533	-282.10	.82	1220.08
4	5	4	4	3	-.434	.007	.019	.188	.074	.507	-.246	.081	.526	-278.12	4.36	1219.29
4	5	4	5	3	-.424	.014	.016	.173	.072	.496	-.251	.087	.512	-271.79	9.21	1214.82
4	5	5	1	3	-.496	-.006	-.117	.244	.072	.585	-.252	.066	.468	-318.03	-4.05	992.71
4	5	5	2	3	-.496	.000	-.114	.246	.067	.585	-.250	.067	.471	-317.86	.14	998.41
4	5	5	3	3	-.492	.007	-.112	.242	.065	.582	-.251	.072	.470	-315.52	4.72	1001.94
4	5	5	4	3	-.485	.017	-.111	.232	.061	.574	-.253	.078	.463	-310.86	10.66	1003.63
4	5	5	5	3	-.473	.030	-.111	.215	.052	.561	-.259	.082	.449	-303.38	19.10	1003.21
4	6	1	1	3	-.077	-.016	.145	-.258	.199	.178	-.335	.182	.322	-49.29	-10.57	1429.33
4	6	1	2	3	-.074	-.019	.144	-.267	.200	.182	-.340	.181	.326	-47.23	-11.90	1429.13
4	6	1	3	3	-.069	-.021	.146	-.281	.198	.186	-.350	.177	.332	-44.08	-13.49	1431.64
4	6	1	4	3	-.063	-.022	.157	-.304	.183	.187	-.368	.161	.344	-40.59	-14.05	1449.77
4	6	1	5	3	-.060	-.015	.251	-.350	.115	.144	-.410	.100	.395	-38.64	-9.93	1606.03
4	6	2	1	3	-.172	-.012	.139	-.117	.191	.265	-.289	.179	.404	-110.33	-7.43	1419.28
4	6	2	2	3	-.164	-.011	.139	-.131	.186	.268	-.296	.175	.407	-105.27	-7.06	1420.26
4	6	2	3	3	-.152	-.010	.142	-.155	.176	.270	-.306	.165	.413	-97.36	-6.64	1425.67
4	6	2	4	3	-.136	-.007	.159	-.188	.147	.266	-.324	.140	.426	-87.38	-4.24	1453.60
4	6	2	5	3	-.124	.009	.281	-.228	.055	.205	-.353	.064	.486	-79.63	5.67	1656.09

NO.OF SEAM-- 1

BLOCK COL	ROW	ELEM COL	ROW	CODE	RIDE1	RIDE2	CLOSURE	U1POS	U2POS	U3POS	U1NEG	U2NEG	U3NEG	SIGS1	SIGS2	SIGN
4	6	3	1	3	-.297	-.001	.093	.054	.178	.384	-.244	.177	.477	-190.55	-.83	1343.03
4	6	3	2	3	-.283	.005	.095	.032	.162	.385	-.251	.167	.480	-181.40	3.40	1347.40
4	6	3	3	3	-.261	.014	.102	-.002	.140	.383	-.263	.154	.485	-167.04	8.76	1358.51
4	6	3	4	3	-.231	.028	.125	-.050	.097	.373	-.280	.125	.498	-147.83	17.67	1396.62
4	6	3	5	3	-.203	.058	.257	-.103	-.007	.300	-.307	.050	.557	-130.38	36.95	1616.02
4	6	4	1	3	-.414	.015	-.006	.201	.160	.508	-.214	.174	.503	-265.55	9.47	1178.95
4	6	4	2	3	-.394	.031	.001	.172	.129	.507	-.221	.160	.508	-252.35	19.88	1189.99
4	6	4	3	3	-.361	.052	.014	.127	.090	.502	-.234	.142	.516	-231.69	33.36	1211.90
4	6	4	4	3	-.317	.082	.047	.065	.029	.487	-.252	.112	.534	-203.43	52.78	1267.13
4	6	4	5	3	-.274	.134	.190	-.005	-.090	.403	-.279	.044	.594	-175.90	85.63	1505.73
4	6	5	1	3	-.460	.037	-.131	.237	.136	.575	-.223	.173	.445	-295.18	23.84	970.52
4	6	5	2	3	-.436	.066	-.119	.206	.088	.574	-.230	.154	.455	-279.65	42.27	990.54
4	6	5	3	3	-.398	.103	-.097	.157	.029	.570	-.241	.132	.473	-255.38	66.07	1027.52
4	6	5	4	3	-.347	.154	-.049	.089	-.053	.555	-.258	.101	.506	-222.32	98.54	1107.00
4	6	5	5	3	-.296	.230	.110	.013	-.187	.472	-.283	.042	.582	-189.82	147.13	1371.02
4	7	1	1	1	-.111	.002	2.214	-.306	.261	-.855	-.416	.263	1.359	.00	.00	.00
4	7	1	2	3	-.035	.013	.225	-.295	.372	.113	-.331	.385	.338	-22.63	8.64	1562.80
4	7	1	3	3	-.009	.012	.126	-.287	.314	.123	-.296	.326	.249	-5.93	7.60	1399.19
4	7	1	4	3	.007	.003	.109	-.293	.312	.093	-.286	.314	.203	4.66	1.68	1370.68
4	7	1	5	3	.016	-.006	.098	-.301	.323	.067	-.285	.317	.165	10.28	-4.01	1351.63
4	7	2	1	1	-.212	.060	2.440	-.140	.215	-.900	-.352	.276	1.540	.01	.00	.01
4	7	2	2	3	-.070	.047	.251	-.226	.369	.157	-.296	.416	.407	-44.85	30.17	1606.36
4	7	2	3	3	-.027	.038	.127	-.258	.300	.163	-.285	.337	.290	-17.08	24.08	1400.65
4	7	2	4	3	-.001	.019	.106	-.281	.298	.122	-.282	.317	.228	-.67	12.27	1365.84
4	7	2	5	3	.013	.002	.095	-.296	.314	.086	-.283	.316	.181	8.02	1.54	1347.22
4	7	3	1	1	-.326	.162	2.404	-.008	.133	-.808	-.333	.294	1.596	.00	-.00	.00
4	7	3	2	3	-.108	.098	.245	-.159	.318	.222	-.266	.416	.467	-69.01	62.97	1596.28
4	7	3	3	3	-.045	.073	.126	-.223	.263	.207	-.268	.336	.333	-29.00	46.86	1398.92
4	7	3	4	3	-.010	.041	.108	-.263	.274	.149	-.273	.315	.257	-6.63	26.34	1368.56
4	7	3	5	3	.008	.014	.098	-.285	.300	.103	-.277	.314	.201	5.17	8.81	1351.24
4	7	4	1	1	-.422	.311	2.254	.090	.011	-.666	-.332	.322	1.588	-.00	-.00	.01
4	7	4	2	3	-.139	.169	.218	-.110	.235	.296	-.249	.404	.514	-88.85	108.31	1551.82
4	7	4	3	3	-.060	.119	.119	-.196	.207	.254	-.255	.326	.373	-38.26	76.42	1386.78
4	7	4	4	3	-.017	.068	.108	-.247	.240	.177	-.264	.308	.286	-11.04	43.76	1368.95
4	7	4	5	3	.005	.027	.100	-.275	.282	.119	-.270	.310	.219	3.07	17.59	1355.21
4	7	5	1	1	-.449	.492	2.063	.110	-.133	-.529	-.339	.359	1.534	-.00	.00	-.00
4	7	5	2	3	-.146	.252	.183	-.099	.136	.361	-.245	.388	.543	-93.48	161.57	1493.20
4	7	5	3	3	-.062	.171	.109	-.188	.142	.294	-.250	.313	.403	-39.90	109.62	1370.45
4	7	5	4	3	-.018	.097	.109	-.241	.202	.201	-.259	.299	.309	-11.70	62.46	1369.39
4	7	5	5	3	.004	.042	.103	-.270	.263	.133	-.265	.305	.236	2.77	26.73	1360.46
5	2	1	1	3	.007	-.043	.136	-.239	-.343	.117	-.232	-.385	.254	4.52	-27.26	1415.73
5	2	1	2	3	-.013	-.109	.144	-.213	-.263	.183	-.226	-.372	.327	-8.28	-69.90	1428.31
5	2	1	3	3	-.052	-.203	.143	-.163	-.174	.274	-.215	-.377	.417	-33.44	-129.99	1426.42
5	2	1	4	3	-.127	-.318	.207	-.079	-.132	.338	-.206	-.449	.545	-81.68	-203.65	1533.03
5	2	1	5	1	-.399	-.663	2.087	.113	.197	-.575	-.286	-.465	1.512	.02	.01	.00
5	2	2	1	3	.010	-.054	.142	-.232	-.327	.123	-.222	-.380	.264	6.17	-34.43	1424.74
5	2	2	2	3	-.006	-.132	.148	-.210	-.230	.191	-.216	-.362	.339	-3.83	-84.68	1435.38
5	2	2	3	3	-.037	-.245	.141	-.168	-.117	.284	-.205	-.362	.425	-23.75	-156.82	1423.05
5	2	2	4	3	-.097	-.388	.188	-.098	-.045	.351	-.195	-.433	.539	-62.30	-248.39	1501.63
5	2	2	5	1	-.312	-.822	1.971	.061	.328	-.538	-.251	-.494	1.432	-.01	-.00	.02
5	2	3	1	3	.013	-.062	.145	-.228	-.314	.124	-.215	-.376	.269	8.27	-39.91	1430.53
5	2	3	2	3	.003	-.149	.151	-.212	-.204	.190	-.209	-.353	.341	2.17	-95.46	1440.24
5	2	3	3	3	-.017	-.275	.139	-.182	-.072	.280	-.199	-.347	.420	-10.76	-176.12	1420.90
5	2	3	4	3	-.058	-.439	.174	-.128	.024	.345	-.187	-.415	.519	-37.40	-281.36	1477.82
5	2	3	5	1	-.206	-.942	1.875	-.012	.432	-.524	-.218	-.510	1.351	-.00	-.01	-.04

NO.OF SEAM-- 1

BLOCK ELEM

COL	ROW	COL	ROW	CODE	RIDE1	RIDE2	CLOSURE	U1POS	U2POS	U3POS	U1NEG	U2NEG	U3NEG	SIGS1	SIGS2	SIGN
5	2	4	1	3	.016	-.068	.149	-.226	-.306	.121	-.210	-.374	.270	10.11	-43.85	1436.55
5	2	4	2	3	.013	-.158	.155	-.219	-.187	.179	-.206	-.346	.334	8.35	-101.53	1447.27
5	2	4	3	3	.005	-.290	.142	-.200	-.042	.260	-.195	-.331	.401	2.99	-185.81	1424.37
5	2	4	4	3	-.018	-.466	.165	-.160	.076	.318	-.178	-.391	.482	-11.73	-299.00	1462.70
5	2	4	5	1	-.104	-1.013	1.793	-.077	.504	-.528	-.181	-.509	1.265	-.00	.00	.02
5	2	5	1	3	.017	-.082	.161	-.231	-.299	.111	-.213	-.382	.272	11.17	-52.70	1456.52
5	2	5	2	3	.021	-.168	.170	-.233	-.176	.153	-.211	-.344	.323	13.56	-107.98	1472.13
5	2	5	3	3	.023	-.289	.156	-.224	-.023	.216	-.202	-.312	.372	14.62	-185.29	1448.26
5	2	5	4	3	.017	-.454	.158	-.189	.114	.265	-.172	-.340	.424	10.62	-290.89	1452.17
5	2	5	5	1	-.020	-.979	1.618	-.103	.521	-.497	-.123	-.458	1.121	.00	-.00	-.01
5	3	1	1	3	-.251	-.347	.337	.014	.230	.297	-.238	-.118	.635	-161.07	-222.71	1750.75
5	3	1	2	3	-.303	-.261	.150	.093	.099	.379	-.210	-.162	.529	-194.27	-167.61	1438.93
5	3	1	3	3	-.356	-.190	.081	.167	.000	.394	-.190	-.190	.475	-228.48	-121.97	1323.56
5	3	1	4	3	-.396	-.134	.053	.221	-.077	.395	-.175	-.211	.448	-253.94	-86.02	1276.77
5	3	1	5	3	-.422	-.095	.047	.255	-.138	.389	-.166	-.233	.437	-270.46	-61.19	1267.22
5	3	2	1	3	-.201	-.436	.277	-.017	.325	.270	-.217	-.111	.548	-128.58	-279.42	1650.51
5	3	2	2	3	-.244	-.331	.077	.052	.182	.333	-.192	-.149	.410	-156.68	-212.22	1316.78
5	3	2	3	3	-.290	-.242	-.005	.114	.064	.334	-.175	-.179	.329	-185.76	-155.21	1180.80
5	3	2	4	3	-.324	-.171	-.040	.161	-.033	.326	-.163	-.204	.285	-207.64	-109.54	1121.20
5	3	2	5	3	-.347	-.120	-.051	.191	-.111	.316	-.156	-.231	.265	-222.25	-76.91	1104.21
5	3	3	1	3	-.137	-.504	.231	-.072	.401	.231	-.208	-.102	.462	-87.50	-322.76	1573.70
5	3	3	2	3	-.174	-.384	.022	-.017	.247	.280	-.190	-.137	.302	-111.37	-246.42	1224.59
5	3	3	3	3	-.210	-.282	-.068	.031	.113	.271	-.179	-.168	.203	-134.41	-180.54	1075.05
5	3	3	4	3	-.236	-.199	-.109	.066	.001	.256	-.170	-.198	.147	-151.25	-127.56	1006.92
5	3	3	5	3	-.254	-.139	-.123	.089	-.090	.243	-.165	-.229	.120	-162.86	-89.41	984.04
5	3	4	1	3	-.080	-.548	.202	-.120	.455	.188	-.200	-.093	.390	-51.32	-351.52	1525.51
5	3	4	2	3	-.114	-.421	-.012	-.081	.293	.232	-.194	-.128	.221	-72.83	-269.95	1168.75
5	3	4	3	3	-.140	-.309	-.105	-.050	.147	.220	-.190	-.162	.115	-90.06	-197.92	1013.17
5	3	4	4	3	-.158	-.220	-.148	-.029	.026	.203	-.187	-.194	.055	-101.28	-140.71	942.29
5	3	4	5	3	-.171	-.156	-.164	-.014	-.072	.188	-.185	-.228	.024	-109.33	-99.83	915.82
5	3	5	1	3	-.042	-.551	.166	-.145	.467	.153	-.187	-.084	.320	-26.99	-353.00	1465.74
5	3	5	2	3	-.070	-.433	-.035	-.134	.310	.194	-.204	-.123	.159	-44.65	-277.57	1129.36
5	3	5	3	3	-.087	-.323	-.123	-.125	.162	.182	-.212	-.161	.059	-55.51	-206.89	982.73
5	3	5	4	3	-.095	-.237	-.162	-.120	.041	.164	-.216	-.197	.001	-60.92	-152.19	918.15
5	3	5	5	3	-.101	-.175	-.179	-.115	-.057	.146	-.216	-.232	-.033	-64.78	-112.08	890.83
5	4	1	1	3	-.434	-.046	.056	.226	-.075	.378	-.208	-.121	.434	-278.19	-29.59	1280.94
5	4	1	2	3	-.447	-.033	.042	.244	-.080	.398	-.203	-.113	.440	-286.29	-20.88	1258.07
5	4	1	3	3	-.455	-.020	.035	.255	-.080	.409	-.199	-.100	.444	-291.47	-12.82	1247.40
5	4	1	4	3	-.459	-.011	.037	.260	-.077	.411	-.198	-.088	.448	-294.14	-6.79	1249.57
5	4	1	5	3	-.459	-.007	.044	.260	-.075	.405	-.200	-.082	.449	-294.50	-4.64	1262.27
5	4	2	1	3	-.358	-.063	-.042	.163	-.053	.297	-.195	-.116	.255	-229.32	-40.63	1118.13
5	4	2	2	3	-.370	-.044	-.059	.180	-.065	.317	-.190	-.109	.258	-236.92	-28.20	1090.92
5	4	2	3	3	-.377	-.027	-.067	.191	-.071	.326	-.187	-.099	.260	-241.93	-17.44	1077.56
5	4	2	4	3	-.381	-.014	-.066	.196	-.074	.328	-.186	-.088	.262	-244.51	-9.20	1078.55
5	4	2	5	3	-.382	-.008	-.058	.196	-.078	.323	-.186	-.086	.264	-244.95	-4.81	1090.95
5	4	3	1	3	-.263	-.078	-.114	.063	-.034	.219	-.201	-.112	.105	-168.90	-50.26	998.76
5	4	3	2	3	-.274	-.054	-.132	.078	-.052	.236	-.197	-.107	.104	-175.82	-34.80	968.33
5	4	3	3	3	-.281	-.033	-.141	.087	-.064	.245	-.194	-.097	.104	-180.44	-21.45	952.78
5	4	3	4	3	-.285	-.017	-.142	.092	-.071	.247	-.193	-.088	.106	-182.74	-11.05	952.41
5	4	3	5	3	-.286	-.007	-.135	.092	-.081	.243	-.194	-.088	.108	-183.16	-4.75	963.78
5	4	4	1	3	-.178	-.092	-.155	-.038	-.017	.160	-.217	-.109	.005	-114.30	-58.66	930.19
5	4	4	2	3	-.188	-.063	-.175	-.026	-.041	.177	-.214	-.104	.002	-120.63	-40.59	896.26
5	4	4	3	3	-.195	-.039	-.186	-.018	-.057	.186	-.212	-.096	.000	-124.70	-24.85	878.71
5	4	4	4	3	-.197	-.020	-.187	-.014	-.069	.188	-.211	-.089	.001	-126.52	-12.64	877.33
5	4	4	5	3	-.198	-.008	-.180	-.014	-.083	.184	-.212	-.090	.004	-126.75	-4.82	887.75

NO.OF SEAM-- 1

BLOCK ELEM

COL	ROW	COL	ROW	CODE	RIDE1	RIDE2	CLOSURE	U1POS	U2POS	U3POS	U1NEG	U2NEG	U3NEG	SIGS1	SIGS2	SIGN
5	4	5	1	3	-.107	-.102	-.173	-.136	-.004	.119	-.244	-.106	-.054	-68.76	-65.18	900.12
5	4	5	2	3	-.116	-.069	-.195	-.127	-.033	.137	-.243	-.102	-.058	-74.53	-44.18	862.72
5	4	5	3	3	-.121	-.043	-.207	-.121	-.052	.146	-.243	-.095	-.060	-77.64	-27.37	844.27
5	4	5	4	3	-.123	-.023	-.208	-.119	-.066	.148	-.242	-.089	-.060	-78.72	-14.68	842.54
5	4	5	5	3	-.122	-.009	-.202	-.119	-.083	.144	-.242	-.092	-.058	-78.49	-5.81	852.04
5	5	1	1	3	-.459	.007	.044	.260	.075	.405	-.200	.081	.450	-294.52	4.40	1262.48
5	5	1	2	3	-.459	.010	.037	.261	.077	.412	-.198	.087	.448	-294.22	6.45	1249.72
5	5	1	3	3	-.455	.019	.035	.255	.080	.410	-.199	.099	.445	-291.61	12.28	1247.39
5	5	1	4	3	-.447	.031	.042	.244	.081	.400	-.203	.112	.442	-286.51	19.99	1257.75
5	5	1	5	3	-.434	.044	.055	.226	.077	.381	-.208	.121	.436	-278.52	28.14	1279.98
5	5	2	1	3	-.382	.007	-.058	.196	.078	.323	-.186	.085	.265	-244.96	4.55	1091.16
5	5	2	2	3	-.381	.014	-.066	.196	.074	.329	-.185	.087	.263	-244.53	8.83	1078.67
5	5	2	3	3	-.377	.026	-.067	.191	.072	.327	-.187	.098	.261	-241.96	16.87	1077.50
5	5	2	4	3	-.370	.043	-.059	.180	.066	.318	-.190	.109	.259	-236.94	27.28	1090.51
5	5	2	5	3	-.358	.061	-.043	.163	.055	.299	-.195	.116	.257	-229.28	39.16	1117.05
5	5	3	1	3	-.286	.007	-.135	.092	.080	.243	-.194	.087	.108	-183.16	4.48	963.98
5	5	3	2	3	-.285	.017	-.142	.092	.071	.248	-.193	.088	.106	-182.72	10.70	952.52
5	5	3	3	3	-.281	.033	-.141	.087	.064	.246	-.194	.096	.105	-180.38	20.91	952.71
5	5	3	4	3	-.274	.053	-.132	.077	.053	.237	-.197	.106	.105	-175.67	33.97	967.94
5	5	3	5	3	-.263	.076	-.114	.062	.035	.220	-.201	.112	.106	-168.57	48.98	997.80
5	5	4	1	3	-.198	.007	-.180	-.014	.082	.184	-.212	.089	.004	-126.73	4.58	887.95
5	5	4	2	3	-.197	.019	-.187	-.014	.069	.188	-.211	.088	.002	-126.46	12.32	877.44
5	5	4	3	3	-.194	.038	-.186	-.018	.057	.187	-.212	.095	.001	-124.58	24.39	878.68
5	5	4	4	3	-.188	.062	-.175	-.026	.041	.178	-.214	.103	.003	-120.37	39.91	895.98
5	5	4	5	3	-.178	.090	-.155	-.039	.018	.161	-.217	.108	.006	-113.81	57.66	929.49
5	5	5	1	3	-.122	.009	-.202	-.119	.082	.144	-.242	.091	-.058	-78.47	5.60	852.22
5	5	5	2	3	-.123	.023	-.207	-.119	.066	.148	-.242	.088	-.059	-78.64	14.42	842.65
5	5	5	3	3	-.121	.042	-.206	-.122	.052	.147	-.243	.094	-.059	-77.50	27.02	844.28
5	5	5	4	3	-.116	.068	-.196	-.127	.033	.138	-.243	.101	-.057	-74.25	43.69	862.57
5	5	5	5	3	-.106	.101	-.173	-.137	.005	.120	-.244	.105	-.054	-68.27	64.50	899.68
5	6	1	1	3	-.423	.091	.046	.256	.144	.397	-.166	.235	.443	-270.92	58.39	1265.56
5	6	1	2	3	-.397	.127	.050	.222	.086	.406	-.175	.214	.457	-254.53	81.70	1272.45
5	6	1	3	3	-.358	.181	.075	.168	.012	.413	-.189	.193	.488	-229.18	116.11	1314.04
5	6	1	4	3	-.304	.252	.139	.095	-.086	.408	-.209	.166	.548	-195.04	161.43	1420.67
5	6	1	5	3	-.252	.343	.319	.015	-.221	.338	-.237	.122	.657	-161.83	219.56	1719.62
5	6	2	1	3	-.346	.116	-.052	.190	.117	.322	-.156	.233	.270	-221.97	74.20	1102.45
5	6	2	2	3	-.323	.165	-.043	.159	.042	.336	-.164	.206	.293	-206.91	105.49	1116.95
5	6	2	3	3	-.287	.234	-.010	.111	-.053	.350	-.176	.181	.340	-184.17	149.90	1171.88
5	6	2	4	3	-.240	.323	.067	.046	-.170	.359	-.194	.152	.426	-153.70	206.84	1300.35
5	6	2	5	3	-.193	.432	.261	-.026	-.317	.306	-.219	.115	.567	-123.78	276.93	1623.27
5	6	3	1	3	-.253	.136	-.123	.087	.094	.248	-.166	.230	.125	-162.05	87.16	982.61
5	6	3	2	3	-.233	.194	-.111	.062	.005	.265	-.171	.199	.154	-149.67	124.36	1003.62
5	6	3	3	3	-.205	.276	-.072	.025	-.105	.284	-.180	.171	.212	-131.52	176.62	1068.49
5	6	3	4	3	-.166	.379	.015	-.027	-.239	.300	-.193	.139	.315	-106.61	242.78	1213.21
5	6	3	5	3	-.125	.501	.220	-.086	-.397	.258	-.211	.104	.478	-80.41	321.44	1555.71
5	6	4	1	3	-.169	.153	-.164	-.016	.076	.192	-.185	.229	.028	-108.31	98.18	914.93
5	6	4	2	3	-.155	.216	-.149	-.033	-.021	.209	-.188	.195	.060	-99.48	138.53	940.22
5	6	4	3	3	-.136	.305	-.107	-.056	-.142	.230	-.192	.163	.122	-87.06	195.45	1009.26
5	6	4	4	3	-.107	.418	-.016	-.090	-.289	.246	-.197	.129	.230	-68.28	267.93	1162.36
5	6	4	5	3	-.070	.548	.197	-.133	-.454	.205	-.203	.094	.402	-44.92	351.10	1515.97
5	6	5	1	3	-.100	.173	-.179	-.117	.059	.149	-.216	.232	-.030	-63.82	111.04	890.41
5	6	5	2	3	-.093	.235	-.163	-.124	-.038	.168	-.217	.197	.005	-59.36	150.92	917.14
5	6	5	3	3	-.083	.321	-.125	-.131	-.159	.189	-.214	.162	.064	-53.15	205.57	980.86
5	6	5	4	3	-.064	.432	-.037	-.142	-.308	.203	-.206	.123	.166	-41.32	276.63	1126.48
5	6	5	5	3	-.035	.551	.164	-.154	-.467	.164	-.190	.084	.328	-22.59	353.01	1461.73

NO.OF SEAM-- 1

BLOCK COL	BLOCK ROW	ELEM COL	ELEM ROW	CODE	RIDE1	RIDE2	CLOSURE	U1POS	U2POS	U3POS	U1NEG	U2NEG	U3NEG	SIGS1	SIGS2	SIGN
5	7	1	1	1	-.401	.670	2.022	.115	-.204	-.507	-.286	.466	1.515	-.00	-.00	-.00
5	7	1	2	3	-.128	.329	.191	-.078	.114	.378	-.206	.444	.569	-82.11	210.93	1506.89
5	7	1	3	3	-.052	.215	.136	-.163	.157	.301	-.215	.372	.437	-33.65	137.56	1415.04
5	7	1	4	3	-.013	.118	.142	-.213	.250	.200	-.226	.368	.342	-8.38	75.33	1424.72
5	7	1	5	3	.007	.047	.136	-.239	.335	.127	-.232	.382	.263	4.49	30.41	1415.45
5	7	2	1	1	-.296	.829	1.914	.046	-.335	-.478	-.250	.495	1.436	.00	.00	-.00
5	7	2	2	3	-.090	.398	.174	-.106	.030	.387	-.197	.427	.561	-57.82	255.04	1479.20
5	7	2	3	3	-.033	.255	.135	-.174	.102	.309	-.207	.357	.444	-21.18	163.67	1413.35
5	7	2	4	3	-.004	.140	.146	-.213	.218	.207	-.217	.358	.353	-2.53	89.66	1432.41
5	7	2	5	3	.011	.058	.142	-.233	.320	.132	-.223	.378	.274	6.75	37.40	1424.63
5	7	3	1	1	-.182	.948	1.838	-.034	-.438	-.482	-.216	.510	1.356	.00	.00	.01
5	7	3	2	3	-.048	.447	.165	-.141	-.036	.372	-.189	.411	.537	-30.98	286.30	1463.61
5	7	3	3	3	-.011	.283	.136	-.190	.061	.300	-.201	.343	.436	-6.99	181.20	1414.95
5	7	3	4	3	.006	.155	.150	-.217	.195	.203	-.210	.350	.353	4.15	99.26	1438.66
5	7	3	5	3	.014	.066	.145	-.230	.308	.132	-.216	.374	.278	9.20	42.28	1430.81
5	7	4	1	1	-.083	1.018	1.774	-.096	-.509	-.502	-.180	.509	1.273	-.00	-.01	-.01
5	7	4	2	3	-.009	.471	.160	-.171	-.084	.336	-.181	.388	.496	-6.05	302.23	1455.77
5	7	4	3	3	.010	.295	.140	-.208	.034	.273	-.198	.329	.413	6.46	189.07	1421.81
5	7	4	4	3	.016	.162	.155	-.223	.181	.188	-.207	.344	.344	10.30	104.06	1446.99
5	7	4	5	3	.017	.071	.149	-.228	.302	.127	-.211	.373	.276	11.08	45.56	1437.20
5	7	5	1	1	-.006	.982	1.611	-.117	-.524	-.482	-.123	.458	1.129	.00	.00	.00
5	7	5	2	3	.023	.457	.157	-.197	-.119	.277	-.175	.338	.434	14.57	292.80	1449.75
5	7	5	3	3	.027	.292	.156	-.230	.018	.225	-.203	.310	.380	17.20	187.15	1447.77
5	7	5	4	3	.024	.171	.171	-.236	.172	.159	-.212	.343	.330	15.17	109.47	1472.62
5	7	5	5	3	.019	.084	.161	-.233	.297	.115	-.214	.381	.277	12.07	53.78	1457.35

NO.OF SEAM-- 2

BLOCK COL	BLOCK ROW	ELEM COL	ELEM ROW	CODE	RIDE1	RIDE2	CLOSURE	U1POS	U2POS	U3POS	U1NEG	U2NEG	U3NEG	SIGS1	SIGS2	SIGN
4	2	1	1	3	.003	.012	.067	-.509	-.543	-.118	-.506	-.531	-.052	2.55	9.84	1239.46
4	2	1	2	3	.006	.016	.075	-.535	-.553	-.143	-.529	-.537	-.067	4.73	12.52	1257.39
4	2	1	3	3	.010	.017	.091	-.575	-.574	-.173	-.565	-.557	-.082	8.38	13.86	1289.61
4	2	1	4	3	.017	.012	.176	-.646	-.635	-.234	-.629	-.623	-.058	13.42	9.63	1466.10
4	2	1	5	1	.041	-.004	2.073	-.756	-.509	-1.189	-.715	-.512	.884	-.00	-.00	-.04
4	2	2	1	3	.003	.015	.067	-.516	-.565	-.124	-.513	-.551	-.057	2.34	11.93	1239.56
4	2	2	2	3	.005	.020	.078	-.548	-.588	-.151	-.543	-.568	-.073	4.13	15.98	1262.82
4	2	2	3	3	.009	.023	.102	-.595	-.628	-.189	-.587	-.605	-.087	6.91	18.20	1313.44
4	2	2	4	3	.013	.016	.222	-.666	-.718	-.271	-.653	-.702	-.049	10.57	12.72	1562.72
4	2	2	5	1	.032	-.005	2.433	-.756	-.554	-1.383	-.725	-.559	1.050	-.00	-.00	-.01
4	2	3	1	3	.003	.016	.071	-.516	-.593	-.130	-.513	-.577	-.059	2.24	13.09	1248.79
4	2	3	2	3	.004	.022	.087	-.552	-.630	-.161	-.548	-.608	-.074	3.29	17.72	1281.14
4	2	3	3	3	.006	.025	.121	-.609	-.691	-.204	-.604	-.666	-.083	4.47	20.03	1352.91
4	2	3	4	3	.007	.017	.271	-.699	-.811	-.301	-.693	-.794	-.031	5.51	13.35	1664.38
4	2	3	5	1	.014	-.009	2.743	-.828	-.634	-1.543	-.815	-.643	1.200	-.00	-.00	.01
4	2	4	1	3	.003	.017	.077	-.508	-.623	-.135	-.506	-.606	-.059	2.03	13.69	1259.85
4	2	4	2	3	.003	.023	.098	-.547	-.677	-.169	-.544	-.655	-.071	2.42	18.21	1304.23
4	2	4	3	3	.003	.025	.147	-.613	-.769	-.218	-.610	-.744	-.072	2.31	19.90	1405.77
4	2	4	4	3	.002	.015	.345	-.732	-.947	-.338	-.730	-.932	.008	1.23	11.71	1819.60
4	2	4	5	1	-.002	-.018	3.337	-.949	-.801	-1.835	-.952	-.820	1.501	-.00	-.00	.04
4	2	5	1	3	.002	.017	.083	-.494	-.653	-.140	-.492	-.636	-.057	1.70	13.91	1272.80
4	2	5	2	3	.002	.022	.111	-.530	-.727	-.175	-.528	-.705	-.065	1.68	17.87	1331.06
4	2	5	3	3	.001	.023	.177	-.595	-.856	-.231	-.594	-.833	-.054	.86	18.60	1469.74
4	2	5	4	3	-.002	.011	.449	-.721	-1.127	-.383	-.723	-1.116	.066	-1.36	8.68	2035.90
4	2	5	5	1	-.012	-.034	4.458	-.968	-1.122	-2.385	-.980	-1.156	2.073	-.00	-.00	-.01
4	3	1	1	3	.017	-.014	.207	-.986	-.394	-.244	-.969	-.408	-.037	13.41	-11.35	1532.01
4	3	1	2	3	.011	-.017	.136	-.996	-.436	-.188	-.985	-.452	-.052	8.74	-13.41	1383.68
4	3	1	3	3	.007	-.012	.132	-1.021	-.425	-.167	-1.014	-.437	-.035	5.49	-9.97	1374.46
4	3	1	4	3	.005	-.007	.133	-1.042	-.396	-.153	-1.037	-.403	-.020	3.67	-5.76	1377.34
4	3	1	5	3	.004	-.003	.134	-1.054	-.360	-.145	-1.050	-.364	-.010	2.91	-2.76	1380.02
4	3	2	1	3	.011	-.019	.279	-1.052	-.415	-.291	-1.041	-.434	-.013	8.98	-14.99	1680.98
4	3	2	2	3	.006	-.022	.189	-1.103	-.468	-.221	-1.097	-.490	-.032	4.49	-17.79	1493.35
4	3	2	3	3	.002	-.016	.186	-1.148	-.454	-.195	-1.147	-.470	-.009	1.42	-13.21	1487.15
4	3	2	4	3	-.000	-.009	.190	-1.183	-.416	-.180	-1.183	-.425	.010	-.32	-7.54	1496.30
4	3	2	5	3	-.001	-.004	.193	-1.203	-.368	-.170	-1.204	-.372	.023	-1.06	-3.47	1503.38
4	3	3	1	3	.002	-.023	.383	-1.186	-.489	-.344	-1.184	-.512	.039	1.82	-18.48	1898.22
4	3	3	2	3	-.002	-.027	.300	-1.288	-.534	-.273	-1.290	-.561	.027	-1.56	-21.53	1724.94
4	3	3	3	3	-.005	-.020	.308	-1.366	-.503	-.248	-1.371	-.523	.060	-3.88	-16.15	1741.93
4	3	3	4	3	-.007	-.012	.320	-1.419	-.447	-.234	-1.426	-.459	.086	-5.24	-9.36	1767.58
4	3	3	5	3	-.007	-.005	.329	-1.451	-.381	-.225	-1.458	-.387	.103	-5.85	-4.37	1785.30
4	3	4	1	3	-.006	-.031	.718	-1.487	-.651	-.502	-1.493	-.681	.216	-4.76	-24.51	2596.22
4	3	4	2	3	-.009	-.035	.702	-1.685	-.648	-.460	-1.694	-.683	.242	-7.24	-27.93	2563.74
4	3	4	3	3	-.011	-.027	.749	-1.811	-.579	-.453	-1.822	-.606	.297	-8.95	-21.56	2661.45
4	3	4	4	3	-.012	-.016	.786	-1.892	-.495	-.448	-1.905	-.511	.337	-10.00	-13.10	2736.89
4	3	4	5	3	-.013	-.008	.810	-1.943	-.405	-.446	-1.956	-.413	.364	-10.54	-6.49	2787.53
4	3	5	1	1	-.022	-.084	5.634	-1.551	-.948	-2.944	-1.573	-1.031	2.691	.00	-.00	-.02
4	3	5	2	1	-.028	-.096	6.244	-1.782	-.793	-3.214	-1.809	-.889	3.030	-.00	-.00	-.04
4	3	5	3	1	-.031	-.077	6.654	-1.924	-.666	-3.386	-1.955	-.743	3.268	-.00	-.00	-.03
4	3	5	4	1	-.033	-.049	6.930	-2.017	-.549	-3.500	-2.050	-.599	3.430	-.01	-.00	.04
4	3	5	5	1	-.035	-.026	7.118	-2.075	-.436	-3.579	-2.110	-.463	3.540	-.01	-.00	.01
4	4	1	1	3	.003	.000	.148	-1.236	-.240	-.138	-1.233	-.239	.010	2.67	.29	1408.39
4	4	1	2	3	.003	.001	.151	-1.251	-.203	-.141	-1.248	-.203	.010	2.78	.57	1414.38
4	4	1	3	3	.004	.001	.152	-1.259	-.166	-.142	-1.255	-.165	.010	2.93	.46	1416.93
4	4	1	4	3	.004	.000	.152	-1.259	-.129	-.143	-1.255	-.128	.009	3.07	.22	1416.91
4	4	1	5	3	.004	-.000	.151	-1.252	-.091	-.142	-1.248	-.091	.009	3.17	-.11	1414.59

NO.OF SEAM-- 2

BLOCK COL	BLOCK ROW	ELEM COL	ELEM ROW	CODE	RIDE1	RIDE2	CLOSURE	U1POS	U2POS	U3POS	U1NEG	U2NEG	U3NEG	SIGS1	SIGS2	SIGN
4	4	2	1	3	-.002	.000	.210	-1.393	-.261	-.163	-1.395	-.261	.047	-1.33	.11	1538.23
4	4	2	2	3	-.002	.001	.215	-1.412	-.216	-.167	-1.414	-.215	.048	-1.31	.58	1547.44
4	4	2	3	3	-.001	.001	.217	-1.422	-.170	-.169	-1.423	-.169	.048	-1.19	.53	1551.68
4	4	2	4	3	-.001	.000	.217	-1.423	-.124	-.169	-1.424	-.124	.048	-1.04	.29	1552.09
4	4	2	5	3	-.001	-.000	.215	-1.414	-.077	-.168	-1.416	-.077	.047	-.90	-.05	1549.08
4	4	3	1	3	-.008	-.000	.351	-1.649	-.288	-.219	-1.657	-.289	.131	-6.15	-.39	1831.37
4	4	3	2	3	-.008	.001	.358	-1.673	-.232	-.224	-1.681	-.231	.135	-6.27	.41	1846.50
4	4	3	3	3	-.008	.001	.362	-1.686	-.176	-.226	-1.694	-.175	.136	-6.24	.54	1854.00
4	4	3	4	3	-.008	.000	.363	-1.688	-.120	-.227	-1.696	-.119	.136	-6.11	.39	1855.67
4	4	3	5	3	-.007	.000	.361	-1.680	-.062	-.226	-1.688	-.062	.135	-5.94	.08	1852.20
4	4	4	1	3	-.014	-.002	.846	-2.150	-.323	-.445	-2.164	-.325	.401	-10.98	-1.69	2862.57
4	4	4	2	3	-.014	-.000	.861	-2.182	-.252	-.452	-2.196	-.252	.409	-11.29	-.12	2893.97
4	4	4	3	3	-.014	.001	.869	-2.199	-.184	-.456	-2.213	-.184	.413	-11.36	.46	2910.47
4	4	4	4	3	-.014	.001	.872	-2.204	-.117	-.458	-2.218	-.116	.414	-11.26	.53	2916.48
4	4	4	5	3	-.014	.000	.870	-2.197	-.046	-.457	-2.211	-.046	.413	-11.07	.27	2913.63
4	4	5	1	1	-.036	-.010	7.316	-2.278	-.363	-3.657	-2.314	-.373	3.659	-.01	-.00	-.00
4	4	5	2	1	-.037	-.003	7.412	-2.309	-.276	-3.703	-2.346	-.279	3.709	-.00	-.00	-.02
4	4	5	3	1	-.037	.001	7.463	-2.326	-.196	-3.728	-2.363	-.195	3.735	.00	-.00	-.03
4	4	5	4	1	-.037	.002	7.489	-2.333	-.116	-3.741	-2.370	-.115	3.748	.00	-.00	.01
4	4	5	5	1	-.036	.001	7.493	-2.330	-.033	-3.743	-2.366	-.031	3.750	.00	-.00	-.01
4	5	1	1	3	.004	.000	.151	-1.252	.088	-.142	-1.248	.088	.009	3.17	.12	1414.74
4	5	1	2	3	.004	-.000	.152	-1.260	.125	-.143	-1.256	.125	.009	3.07	-.20	1417.14
4	5	1	3	3	.004	-.001	.152	-1.260	.163	-.142	-1.256	.162	.010	2.94	-.45	1417.23
4	5	1	4	3	.003	-.001	.151	-1.253	.200	-.141	-1.249	.199	.010	2.79	-.57	1414.79
4	5	1	5	3	.003	-.000	.148	-1.238	.237	-.138	-1.235	.236	.010	2.68	-.31	1408.96
4	5	2	1	3	-.001	.000	.215	-1.415	.074	-.168	-1.416	.074	.047	-.90	.06	1549.28
4	5	2	2	3	-.001	-.000	.217	-1.423	.121	-.169	-1.425	.120	.048	-1.04	-.28	1552.40
4	5	2	3	3	-.001	-.001	.217	-1.423	.167	-.169	-1.424	.166	.048	-1.19	-.53	1552.11
4	5	2	4	3	-.002	-.001	.215	-1.413	.212	-.167	-1.415	.211	.048	-1.31	-.59	1548.06
4	5	2	5	3	-.002	-.000	.211	-1.395	.258	-.164	-1.397	.257	.047	-1.35	-.15	1539.14
4	5	3	1	3	-.007	-.000	.361	-1.681	.059	-.226	-1.688	.059	.135	-5.94	-.07	1852.44
4	5	3	2	3	-.008	-.000	.363	-1.689	.116	-.227	-1.697	.116	.136	-6.11	-.39	1856.11
4	5	3	3	3	-.008	-.001	.362	-1.687	.172	-.226	-1.695	.171	.136	-6.24	-.56	1854.68
4	5	3	4	3	-.008	-.001	.359	-1.675	.228	-.224	-1.683	.227	.135	-6.29	-.46	1847.52
4	5	3	5	3	-.008	.000	.352	-1.652	.284	-.220	-1.659	.284	.132	-6.20	.29	1832.98
4	5	4	1	3	-.014	-.000	.871	-2.198	.043	-.457	-2.211	.042	.414	-11.08	-.29	2914.01
4	5	4	2	3	-.014	-.001	.872	-2.205	.113	-.458	-2.219	.112	.414	-11.27	-.56	2917.28
4	5	4	3	3	-.014	-.001	.869	-2.201	.180	-.456	-2.215	.179	.413	-11.38	-.52	2911.78
4	5	4	4	3	-.014	-.000	.862	-2.184	.248	-.452	-2.198	.248	.409	-11.32	-.01	2896.05
4	5	4	5	3	-.014	.002	.847	-2.153	.318	-.446	-2.167	.320	.402	-11.05	1.46	2866.00
4	5	5	1	1	-.036	-.001	7.494	-2.330	.029	-3.743	-2.367	.027	3.751	.00	.00	.01
4	5	5	2	1	-.037	-.002	7.491	-2.334	.112	-3.742	-2.371	.110	3.750	.01	.00	-.01
4	5	5	3	1	-.037	-.001	7.467	-2.327	.191	-3.730	-2.365	.190	3.737	.01	.00	-.05
4	5	5	4	1	-.037	.002	7.418	-2.311	.271	-3.706	-2.348	.273	3.712	.01	.01	.00
4	5	5	5	1	-.036	.009	7.327	-2.280	.357	-3.662	-2.316	.366	3.665	.02	.00	-.09
4	6	1	1	3	.004	.003	.135	-1.058	.357	-.145	-1.054	.360	-.010	2.88	2.72	1381.26
4	6	1	2	3	.004	.007	.134	-1.046	.393	-.153	-1.042	.400	-.019	3.59	5.68	1379.06
4	6	1	3	3	.007	.012	.133	-1.026	.421	-.166	-1.020	.434	-.034	5.34	9.87	1376.77
4	6	1	4	3	.011	.017	.137	-1.003	.433	-.188	-.993	.449	-.051	8.52	13.30	1386.73
4	6	1	5	3	.016	.014	.209	-.994	.392	-.244	-.977	.406	-.035	13.09	11.29	1536.27
4	6	2	1	3	-.001	.004	.194	-1.207	.363	-.170	-1.208	.367	.024	-1.14	3.36	1505.38
4	6	2	2	3	-.001	.009	.191	-1.188	.411	-.180	-1.189	.420	.012	-.48	7.36	1499.24
4	6	2	3	3	.001	.016	.188	-1.156	.449	-.195	-1.155	.465	-.007	1.15	12.98	1491.40
4	6	2	4	3	.005	.022	.192	-1.113	.463	-.221	-1.108	.485	-.029	4.08	17.56	1499.39
4	6	2	5	3	.010	.019	.283	-1.064	.411	-.292	-1.054	.430	-.009	8.41	14.84	1690.08

NO.OF SEAM-- 2

BLOCK COL	ROW	ELEM COL	ROW	CODE	RIDE1	RIDE2	CLOSURE	U1POS	U2POS	U3POS	U1NEG	U2NEG	U3NEG	SIGS1	SIGS2	SIGN
4	6	3	1	3	-.007	.005	.330	-1.456	.375	-.225	-1.463	.380	.105	-5.97	4.14	1788.64
4	6	3	2	3	-.007	.011	.323	-1.426	.440	-.234	-1.433	.451	.089	-5.47	9.02	1772.85
4	6	3	3	3	-.005	.020	.312	-1.376	.495	-.248	-1.382	.514	.063	-4.25	15.72	1750.18
4	6	3	4	3	-.003	.026	.306	-1.303	.526	-.273	-1.306	.552	.033	-2.10	21.10	1737.71
4	6	3	5	3	.001	.023	.393	-1.206	.482	-.346	-1.205	.505	.047	1.06	18.20	1918.88
4	6	4	1	3	-.013	.008	.813	-1.948	.397	-.446	-1.962	.405	.367	-10.69	6.06	2794.28
4	6	4	2	3	-.013	.016	.791	-1.902	.484	-.449	-1.914	.500	.342	-10.25	12.47	2748.24
4	6	4	3	3	-.012	.026	.759	-1.825	.566	-.455	-1.837	.591	.304	-9.34	20.78	2680.72
4	6	4	4	3	-.010	.034	.719	-1.709	.632	-.465	-1.719	.666	.253	-7.82	27.15	2597.31
4	6	4	5	3	-.007	.030	.746	-1.525	.636	-.512	-1.532	.666	.234	-5.54	24.00	2655.40
4	6	5	1	1	-.035	.024	7.138	-2.080	.426	-3.587	-2.115	.450	3.551	.01	.00	-.00
4	6	5	2	1	-.034	.047	6.964	-2.024	.535	-3.515	-2.058	.582	3.449	.01	.00	-.03
4	6	5	3	1	-.032	.074	6.716	-1.936	.645	-3.414	-1.968	.719	3.302	.01	.00	.05
4	6	5	4	1	-.029	.093	6.362	-1.805	.762	-3.269	-1.834	.855	3.093	.01	-.00	.04
4	6	5	5	1	-.023	.082	5.875	-1.602	.909	-3.060	-1.625	.990	2.816	.00	-.00	.03
4	7	1	1	1	.040	.004	2.081	-.764	.509	-1.192	-.724	.513	.889	.00	.00	.03
4	7	1	2	3	.016	-.012	.177	-.653	.637	-.234	-.637	.625	-.057	13.13	-9.58	1470.09
4	7	1	3	3	.010	-.017	.092	-.581	.577	-.173	-.571	.560	-.081	8.20	-13.79	1292.19
4	7	1	4	3	.006	-.016	.076	-.539	.556	-.143	-.534	.540	-.067	4.63	-12.47	1259.19
4	7	1	5	3	.003	-.012	.067	-.513	.545	-.118	-.510	.533	-.051	2.51	-9.81	1240.74
4	7	2	1	1	.030	.005	2.451	-.769	.555	-1.390	-.739	.560	1.061	.00	.00	-.01
4	7	2	2	3	.013	-.016	.226	-.677	.722	-.271	-.664	.707	-.045	10.05	-12.59	1571.00
4	7	2	3	3	.008	-.022	.105	-.603	.633	-.189	-.595	.610	-.084	6.58	-18.02	1318.18
4	7	2	4	3	.005	-.020	.079	-.553	.593	-.151	-.548	.573	-.072	3.94	-15.83	1265.78
4	7	2	5	3	.003	-.015	.068	-.520	.569	-.124	-.517	.555	-.056	2.25	-11.84	1241.43
4	7	3	1	1	.011	.009	2.786	-.850	.636	-1.562	-.839	.645	1.224	.00	.00	-.03
4	7	3	2	3	.006	-.016	.279	-.716	.820	-.303	-.710	.803	-.024	4.82	-13.09	1681.97
4	7	3	3	3	.005	-.025	.125	-.620	.699	-.205	-.615	.675	-.079	4.03	-19.69	1361.72
4	7	3	4	3	.004	-.022	.089	-.559	.637	-.161	-.555	.615	-.072	3.03	-17.44	1285.94
4	7	3	5	3	.003	-.016	.073	-.520	.599	-.130	-.517	.582	-.057	2.11	-12.92	1251.49
4	7	4	1	1	-.005	.018	3.452	-.992	.807	-1.889	-.997	.825	1.563	.00	-.00	.05
4	7	4	2	3	.001	-.014	.365	-.758	.968	-.344	-.758	.954	.021	.54	-11.33	1860.24
4	7	4	3	3	.002	-.024	.154	-.626	.784	-.220	-.623	.760	-.065	1.87	-19.38	1421.97
4	7	4	4	3	.003	-.022	.101	-.554	.688	-.169	-.551	.666	-.067	2.16	-17.76	1311.70
4	7	4	5	3	.002	-.017	.078	-.512	.631	-.135	-.510	.614	-.057	1.90	-13.42	1263.58
4	7	5	1	1	-.014	.034	4.848	-1.061	1.137	-2.575	-1.075	1.171	2.273	-.00	-.00	-.03
4	7	5	2	3	-.002	-.010	.498	-.758	1.180	-.403	-.761	1.170	.095	-1.80	-8.21	2138.19
4	7	5	3	3	.001	-.022	.191	-.606	.883	-.235	-.606	.861	-.044	.57	-17.92	1497.48
4	7	5	4	3	.002	-.022	.116	-.535	.742	-.176	-.533	.720	-.060	1.52	-17.30	1341.48
4	7	5	5	3	.002	-.017	.085	-.496	.663	-.140	-.494	.646	-.055	1.61	-13.57	1277.49
5	2	1	1	3	.002	.017	.110	-.496	-.935	-.156	-.493	-.918	-.046	1.99	13.68	1329.18
5	2	1	2	3	.002	.021	.147	-.544	-1.031	-.193	-.542	-1.011	-.047	1.76	16.72	1405.63
5	2	1	3	3	.001	.020	.233	-.620	-1.199	-.255	-.619	-1.179	-.022	.53	16.37	1585.85
5	2	1	4	3	-.003	.006	.576	-.747	-1.551	-.437	-.750	-1.545	.139	-2.23	5.10	2300.35
5	2	1	5	1	-.016	-.046	5.471	-.951	-1.621	-2.876	-.967	-1.667	2.595	-.00	-.00	.00
5	2	2	1	3	.002	.017	.117	-.469	-.971	-.162	-.467	-.954	-.045	1.70	13.45	1343.58
5	2	2	2	3	.001	.020	.159	-.508	-1.085	-.201	-.507	-1.066	-.041	1.20	15.70	1432.48
5	2	2	3	3	-.000	.018	.260	-.568	-1.285	-.266	-.568	-1.267	-.006	-.34	14.46	1642.43
5	2	2	4	3	-.004	.003	.651	-.663	-1.696	-.468	-.667	-1.693	.184	-3.32	2.46	2457.45
5	2	2	5	1	-.019	-.053	6.068	-.802	-1.803	-3.163	-.821	-1.856	2.905	.00	.00	.01
5	2	3	1	3	.002	.016	.121	-.438	-.999	-.166	-.436	-.983	-.045	1.53	12.98	1353.02
5	2	3	2	3	.001	.018	.168	-.464	-1.126	-.205	-.463	-1.108	-.037	.69	14.35	1450.31
5	2	3	3	3	-.002	.015	.278	-.502	-1.346	-.271	-.504	-1.331	.006	-1.27	12.33	1679.08
5	2	3	4	3	-.006	.000	.698	-.562	-1.798	-.483	-.568	-1.797	.215	-4.51	.16	2555.00
5	2	3	5	1	-.021	-.057	6.430	-.654	-1.931	-3.333	-.676	-1.988	3.097	-.00	-.00	-.03

NO.OF SEAM-- 2

BLOCK		ELEM															
COL	ROW	COL	ROW	CODE	RIDE1	RIDE2	CLOSURE	U1POS	U2POS	U3POS	U1NEG	U2NEG	U3NEG	SIGS1	SIGS2	SIGN
5	2	4	1	3	.002	.015	.125	-.404	-1.017	-.169	-.402	-1.002	-.045	1.58	12.16	1360.58
5	2	4	2	3	.000	.016	.173	-.416	-1.148	-.207	-.416	-1.133	-.033	.40	12.43	1461.70
5	2	4	3	3	-.003	.012	.285	-.427	-1.378	-.271	-.430	-1.366	.014	-2.06	9.65	1694.58
5	2	4	4	3	-.007	-.003	.713	-.442	-1.854	-.482	-.449	-1.857	.232	-5.58	-2.06	2586.00
5	2	4	5	1	-.022	-.057	6.583	-.488	-2.035	-3.397	-.510	-2.092	3.186	.00	.00	.03
5	2	5	1	3	.002	.013	.137	-.379	-1.026	-.177	-.378	-1.013	-.040	1.48	10.46	1386.80
5	2	5	2	3	.000	.012	.186	-.377	-1.150	-.212	-.376	-1.138	-.026	.25	9.34	1486.83
5	2	5	3	3	-.003	.008	.285	-.355	-1.369	-.266	-.358	-1.362	.019	-2.28	6.02	1694.12
5	2	5	4	3	-.007	-.005	.657	-.287	-1.827	-.444	-.294	-1.832	.213	-5.45	-3.88	2469.10
5	2	5	5	1	-.019	-.051	6.160	-.205	-2.157	-3.174	-.224	-2.208	2.986	.00	.00	-.01
5	3	1	1	3	-.026	-.098	7.697	-1.482	-1.491	-7.801	-1.508	-1.588	-.105	-.49	-1.83	400.85
5	3	1	2	3	-.031	-.111	8.693	-1.687	-1.224	-8.760	-1.718	-1.335	-.067	-.58	-2.07	452.78
5	3	1	3	3	-.034	-.091	9.307	-1.834	-1.014	-9.340	-1.868	-1.105	-.032	-.64	-1.69	484.83
5	3	1	4	3	-.036	-.060	9.732	-1.935	-.829	-9.738	-1.971	-.888	-.006	-.66	-1.12	506.84
5	3	1	5	3	-.036	-.033	10.052	-1.999	-.649	-10.040	-2.035	-.682	.012	-.67	-.61	523.52
5	3	2	1	3	-.027	-.104	8.601	-1.236	-1.686	-8.690	-1.263	-1.790	-.089	-.50	-1.93	447.96
5	3	2	2	3	-.030	-.116	9.851	-1.390	-1.376	-9.901	-1.420	-1.492	-.050	-.56	-2.15	513.12
5	3	2	3	3	-.031	-.095	10.630	-1.508	-1.122	-10.644	-1.539	-1.216	-.014	-.57	-1.76	553.59
5	3	2	4	3	-.031	-.063	11.164	-1.593	-.899	-11.151	-1.623	-.963	.014	-.56	-1.17	581.47
5	3	2	5	3	-.031	-.036	11.561	-1.647	-.687	-11.530	-1.677	-.723	.032	-.57	-.67	602.16
5	3	3	1	3	-.026	-.104	9.159	-1.029	-1.834	-9.233	-1.055	-1.939	-.074	-.49	-1.94	477.04
5	3	3	2	3	-.025	-.114	10.582	-1.151	-1.497	-10.616	-1.176	-1.612	-.034	-.47	-2.13	551.13
5	3	3	3	3	-.022	-.093	11.476	-1.241	-1.208	-11.473	-1.264	-1.302	.003	-.42	-1.73	597.75
5	3	3	4	3	-.020	-.063	12.086	-1.303	-.958	-12.056	-1.323	-1.020	.031	-.38	-1.18	629.50
5	3	3	5	3	-.018	-.035	12.541	-1.346	-.722	-12.491	-1.364	-.757	.049	-.36	-.65	653.15
5	3	4	1	3	-.023	-.099	9.569	-.847	-1.972	-9.629	-.870	-2.071	-.060	-.42	-1.85	498.36
5	3	4	2	3	-.016	-.107	11.157	-.959	-1.594	-11.177	-.976	-1.701	-.020	-.31	-1.98	581.14
5	3	4	3	3	-.010	-.087	12.115	-1.015	-1.268	-12.099	-1.024	-1.355	.017	-.18	-1.63	631.02
5	3	4	4	3	-.005	-.059	12.753	-1.045	-.997	-12.709	-1.050	-1.056	.044	-.09	-1.09	664.22
5	3	4	5	3	-.002	-.033	13.245	-1.074	-.753	-13.183	-1.076	-.786	.062	-.04	-.62	689.85
5	3	5	1	3	-.013	-.086	10.069	-.729	-2.168	-10.117	-.742	-2.253	-.047	-.23	-1.58	524.44
5	3	5	2	3	-.004	-.092	11.840	-.822	-1.655	-11.848	-.826	-1.746	-.008	-.05	-1.70	616.64
5	3	5	3	3	.005	-.076	12.711	-.808	-1.282	-12.684	-.803	-1.358	.027	.11	-1.41	661.99
5	3	5	4	3	.010	-.054	13.259	-.785	-1.008	-13.206	-.775	-1.062	.053	.21	-1.00	690.56
5	3	5	5	3	.014	-.032	13.749	-.793	-.789	-13.679	-.779	-.820	.070	.27	-.58	716.04
5	4	1	1	3	-.037	-.015	10.289	-2.261	-.577	-10.253	-2.297	-.593	.037	-.68	-.28	535.88
5	4	1	2	3	-.038	-.005	10.376	-2.300	-.441	-10.336	-2.338	-.446	.041	-.70	-.10	540.40
5	4	1	3	3	-.038	-.000	10.435	-2.323	-.315	-10.392	-2.360	-.315	.042	-.70	-.00	543.49
5	4	1	4	3	-.037	.002	10.489	-2.330	-.190	-10.446	-2.367	-.188	.043	-.70	.04	546.28
5	4	1	5	3	-.037	.002	10.542	-2.320	-.057	-10.499	-2.357	-.055	.043	-.69	.03	549.08
5	4	2	1	3	-.031	-.018	11.859	-1.890	-.626	-11.801	-1.921	-.643	.058	-.57	-.33	617.65
5	4	2	2	3	-.031	-.007	11.975	-1.928	-.470	-11.913	-1.959	-.476	.063	-.58	-.13	623.73
5	4	2	3	3	-.031	-.000	12.052	-1.948	-.328	-11.987	-1.979	-.328	.065	-.58	-.01	627.69
5	4	2	4	3	-.031	.002	12.119	-1.954	-.187	-12.053	-1.985	-.185	.066	-.57	.03	631.24
5	4	2	5	3	-.030	.002	12.181	-1.945	-.036	-12.115	-1.975	-.034	.066	-.56	.04	634.43
5	4	3	1	3	-.018	-.018	12.895	-1.565	-.669	-12.820	-1.583	-.687	.076	-.35	-.33	671.65
5	4	3	2	3	-.018	-.006	13.045	-1.598	-.497	-12.964	-1.616	-.503	.081	-.35	-.11	679.44
5	4	3	3	3	-.018	.000	13.140	-1.616	-.339	-13.056	-1.634	-.339	.084	-.34	-.01	684.43
5	4	3	4	3	-.017	.003	13.217	-1.621	-.185	-13.131	-1.638	-.182	.085	-.33	.06	688.36
5	4	3	5	3	-.017	.003	13.280	-1.613	-.019	-13.195	-1.630	-.016	.085	-.31	.05	691.66
5	4	4	1	3	-.002	-.017	13.667	-1.270	-.709	-13.578	-1.271	-.726	.088	-.02	-.31	711.79
5	4	4	2	3	-.001	-.005	13.857	-1.300	-.521	-13.762	-1.301	-.526	.095	-.00	-.10	721.71
5	4	4	3	3	-.000	.002	13.967	-1.313	-.349	-13.869	-1.313	-.347	.098	.01	.04	727.46
5	4	4	4	3	.000	.004	14.047	-1.315	-.182	-13.947	-1.315	-.178	.100	.03	.08	731.59
5	4	4	5	3	.001	.004	14.108	-1.309	-.006	-14.009	-1.308	-.003	.099	.04	.07	734.81

NO.OF SEAM-- 2

BLOCK ELEM

COL	ROW	COL	ROW	CODE	RIDE1	RIDE2	CLOSURE	U1POS	U2POS	U3POS	U1NEG	U2NEG	U3NEG	SIGS1	SIGS2	SIGN
5	4	5	1	3	.016	-.013	14.350	-.993	-.756	-14.255	-.977	-.769	.095	.29	-.24	747.40
5	4	5	2	3	.017	-.001	14.591	-1.017	-.539	-14.489	-1.001	-.540	.102	.30	-.02	759.96
5	4	5	3	3	.018	.004	14.700	-1.020	-.353	-14.594	-1.002	-.349	.106	.32	.08	765.62
5	4	5	4	3	.018	.006	14.768	-1.017	-.179	-14.661	-.999	-.173	.107	.33	.12	769.16
5	4	5	5	3	.019	.004	14.819	-1.012	.002	-14.712	-.993	.006	.107	.33	.08	771.82
5	5	1	1	3	-.037	-.002	10.543	-2.321	.052	-10.500	-2.357	.050	.043	-.68	-.04	549.13
5	5	1	2	3	-.037	-.002	10.492	-2.330	.185	-10.448	-2.367	.183	.043	-.70	-.04	546.42
5	5	1	3	3	-.038	-.000	10.440	-2.323	.309	-10.397	-2.361	.309	.043	-.71	-.00	543.73
5	5	1	4	3	-.038	.005	10.384	-2.301	.435	-10.343	-2.339	.439	.041	-.71	.08	540.88
5	5	1	5	3	-.037	.014	10.303	-2.262	.570	-10.265	-2.299	.584	.037	-.71	.26	536.56
5	5	2	1	3	-.030	-.003	12.182	-1.945	.032	-12.116	-1.975	.029	.066	-.56	-.05	634.47
5	5	2	2	3	-.031	-.003	12.122	-1.954	.182	-12.056	-1.985	.180	.066	-.58	-.05	631.37
5	5	2	3	3	-.031	-.000	12.057	-1.949	.322	-11.992	-1.979	.322	.065	-.59	-.01	628.00
5	5	2	4	3	-.031	.006	11.984	-1.928	.464	-11.921	-1.959	.469	.063	-.59	.11	624.17
5	5	2	5	3	-.031	.016	11.873	-1.890	.618	-11.815	-1.921	.635	.058	-.56	.30	618.43
5	5	3	1	3	-.017	-.003	13.281	-1.613	.014	-13.196	-1.630	.011	.085	-.30	-.06	691.73
5	5	3	2	3	-.017	-.003	13.220	-1.621	.180	-13.134	-1.638	.176	.086	-.31	-.07	688.50
5	5	3	3	3	-.018	-.001	13.146	-1.616	.334	-13.061	-1.634	.333	.085	-.32	-.01	684.67
5	5	3	4	3	-.018	.006	13.054	-1.598	.491	-12.972	-1.616	.496	.082	-.31	.10	679.92
5	5	3	5	3	-.018	.017	12.909	-1.564	.662	-12.833	-1.582	.679	.076	-.32	.31	672.35
5	5	4	1	3	.001	-.004	14.110	-1.309	.001	-14.011	-1.308	-.002	.099	.02	-.07	734.86
5	5	4	2	3	.001	-.004	14.050	-1.315	.177	-13.950	-1.314	.173	.100	.00	-.08	731.73
5	5	4	3	3	-.000	-.002	13.972	-1.313	.343	-13.874	-1.313	.341	.099	-.01	-.04	727.79
5	5	4	4	3	-.001	.004	13.864	-1.299	.515	-13.769	-1.300	.519	.095	-.03	.08	722.09
5	5	4	5	3	-.001	.016	13.678	-1.268	.703	-13.589	-1.269	.719	.089	-.04	.29	712.28
5	5	5	1	3	.019	-.005	14.820	-1.012	-.006	-14.714	-.993	-.011	.107	.34	-.08	771.91
5	5	5	2	3	.019	-.006	14.771	-1.017	.174	-14.663	-.998	.168	.108	.34	-.11	769.34
5	5	5	3	3	.018	-.005	14.704	-1.019	.348	-14.598	-1.002	.343	.106	.34	-.09	765.91
5	5	5	4	3	.017	.001	14.598	-1.016	.534	-14.495	-.999	.535	.103	.32	.01	760.20
5	5	5	5	3	.016	.013	14.359	-.991	.750	-14.264	-.974	.763	.096	.31	.24	748.02
5	6	1	1	3	-.036	.031	10.076	-2.001	.637	-10.063	-2.037	.668	.013	-.68	.57	524.93
5	6	1	2	3	-.036	.057	9.774	-1.938	.812	-9.777	-1.973	.869	-.004	-.67	1.05	509.08
5	6	1	3	3	-.034	.087	9.383	-1.837	.990	-9.412	-1.871	1.076	-.029	-.65	1.61	488.66
5	6	1	4	3	-.031	.107	8.846	-1.690	1.183	-8.908	-1.721	1.290	-.063	-.60	1.99	460.70
5	6	1	5	3	-.026	.095	8.104	-1.485	1.404	-8.204	-1.511	1.500	-.100	-.50	1.78	422.09
5	6	2	1	3	-.030	.033	11.587	-1.646	.676	-11.553	-1.676	.709	.033	-.55	.62	603.50
5	6	2	2	3	-.030	.060	11.206	-1.590	.884	-11.190	-1.620	.944	.016	-.55	1.12	583.47
5	6	2	3	3	-.030	.091	10.700	-1.503	1.100	-10.711	-1.533	1.191	-.011	-.55	1.70	557.41
5	6	2	4	3	-.029	.112	9.978	-1.374	1.346	-10.024	-1.403	1.458	-.046	-.52	2.09	519.75
5	6	2	5	3	-.026	.101	8.844	-1.193	1.648	-8.929	-1.219	1.749	-.085	-.45	1.88	460.60
5	6	3	1	3	-.018	.033	12.564	-1.343	.712	-12.513	-1.361	.745	.051	-.32	.62	654.13
5	6	3	2	3	-.019	.060	12.122	-1.298	.945	-12.089	-1.317	1.005	.033	-.34	1.12	631.56
5	6	3	3	3	-.021	.090	11.532	-1.231	1.193	-11.526	-1.252	1.284	.006	-.38	1.68	600.54
5	6	3	4	3	-.023	.111	10.667	-1.132	1.481	-10.697	-1.155	1.592	-.030	-.42	2.06	555.50
5	6	3	5	3	-.023	.102	9.280	-.996	1.823	-9.350	-1.020	1.926	-.070	-.44	1.91	483.32
5	6	4	1	3	-.001	.032	13.264	-1.070	.745	-13.200	-1.071	.777	.064	-.04	.59	691.07
5	6	4	2	3	-.004	.057	12.781	-1.039	.988	-12.735	-1.042	1.045	.046	-.08	1.06	665.54
5	6	4	3	3	-.008	.085	12.154	-1.004	1.258	-12.136	-1.012	1.343	.019	-.15	1.57	633.07
5	6	4	4	3	-.014	.105	11.210	-.944	1.586	-11.227	-.957	1.690	-.017	-.27	1.95	583.90
5	6	4	5	3	-.020	.098	9.632	-.826	1.969	-9.689	-.845	2.067	-.057	-.37	1.81	501.68
5	6	5	1	3	.014	.031	13.763	-.788	.782	-13.692	-.774	.813	.071	.28	.57	716.69
5	6	5	2	3	.011	.053	13.279	-.779	1.001	-13.224	-.768	1.054	.054	.22	.99	691.67
5	6	5	3	3	.006	.075	12.736	-.798	1.276	-12.708	-.792	1.351	.028	.12	1.40	663.31
5	6	5	4	3	-.002	.090	11.871	-.810	1.651	-11.877	-.812	1.741	-.006	-.02	1.68	618.24
5	6	5	5	3	-.011	.085	10.103	-.714	2.167	-10.148	-.725	2.252	-.045	-.20	1.58	526.17

NO.OF SEAM-- 2

BLOCK COL	BLOCK ROW	ELEM COL	ELEM ROW	CODE	RIDE1	RIDE2	CLOSURE	U1POS	U2POS	U3POS	U1NEG	U2NEG	U3NEG	SIGS1	SIGS2	SIGN
5	7	1	1	1	-.016	.046	7.814	-.953	1.641	-7.949	-.969	1.687	-.135	-.00	.00	.02
5	7	1	2	3	-.003	-.006	.674	-.748	1.655	-.481	-.751	1.649	.193	-2.23	-4.60	2504.66
5	7	1	3	3	.001	-.020	.250	-.621	1.234	-.261	-.620	1.215	-.010	.52	-15.63	1621.98
5	7	1	4	3	.002	-.020	.152	-.545	1.049	-.194	-.543	1.029	-.042	1.75	-16.10	1417.61
5	7	1	5	3	.002	-.017	.112	-.496	.945	-.155	-.494	.929	-.043	1.98	-13.30	1334.37
5	7	2	1	1	-.017	.053	6.466	-.713	1.820	-3.358	-.730	1.872	3.109	.01	-.00	-.00
5	7	2	2	3	-.004	-.003	.701	-.628	1.751	-.488	-.631	1.748	.213	-2.86	-2.02	2561.67
5	7	2	3	3	-.000	-.017	.274	-.558	1.312	-.270	-.558	1.295	.004	-.06	-13.78	1670.74
5	7	2	4	3	.002	-.019	.165	-.505	1.101	-.201	-.503	1.082	-.036	1.35	-15.12	1443.16
5	7	2	5	3	.002	-.016	.119	-.467	.981	-.162	-.465	.964	-.042	1.77	-13.10	1348.47
5	7	3	1	1	-.018	.056	6.565	-.614	1.942	-3.396	-.633	1.999	3.169	-.01	.00	-.02
5	7	3	2	3	-.005	.000	.719	-.538	1.822	-.490	-.542	1.822	.230	-3.77	.18	2599.24
5	7	3	3	3	-.001	-.015	.286	-.490	1.363	-.273	-.491	1.349	.013	-.82	-11.81	1696.26
5	7	3	4	3	.001	-.017	.172	-.458	1.137	-.205	-.457	1.120	-.033	.94	-13.91	1458.21
5	7	3	5	3	.002	-.016	.123	-.434	1.007	-.166	-.432	.991	-.043	1.65	-12.70	1357.05
5	7	4	1	1	-.019	.057	6.645	-.466	2.043	-3.425	-.485	2.099	3.220	.00	-.00	.02
5	7	4	2	3	-.006	.003	.723	-.426	1.867	-.484	-.432	1.869	.239	-4.81	2.27	2607.29
5	7	4	3	3	-.002	-.012	.290	-.417	1.388	-.271	-.419	1.377	.019	-1.60	-9.30	1704.50
5	7	4	4	3	.001	-.015	.176	-.410	1.156	-.207	-.409	1.141	-.031	.65	-12.14	1467.02
5	7	4	5	3	.002	-.015	.126	-.401	1.023	-.169	-.399	1.008	-.043	1.70	-11.98	1363.64
5	7	5	1	1	-.017	.050	6.190	-.190	2.161	-3.186	-.207	2.212	3.003	.00	.00	-.02
5	7	5	2	3	-.006	.005	.662	-.276	1.833	-.445	-.282	1.838	.217	-4.89	3.99	2479.93
5	7	5	3	3	-.002	-.007	.288	-.347	1.375	-.266	-.349	1.368	.022	-1.94	-5.84	1699.84
5	7	5	4	3	.001	-.011	.187	-.371	1.155	-.212	-.371	1.144	-.024	.44	-9.19	1490.38
5	7	5	5	3	.002	-.013	.139	-.376	1.030	-.177	-.374	1.017	-.038	1.58	-10.37	1389.11

TIME TAKEN TO OUTPUT THE RESULTS FOR SEAM LOCATIONS -- 28.0860 SECOND

NO.OF SEAM-- 1

BLOCK COL	BLOCK ROW	CODE	RIDE1	RIDE2	CLOSURE	U1POS	U2POS	U3POS	U1NEG	U2NEG	U3NEG	SIGS1	SIGS2	SIGN
1	1	A	.006	.001	.013	-.169	-.112	.010	-.164	-.110	.023	3.56	.85	1209.67
1	2	A	.010	.004	.184	-.213	-.092	-.053	-.203	-.088	.131	5.33	2.11	1196.12
1	3	A	.011	.005	.019	-.243	-.058	.014	-.232	-.053	.033	6.94	3.29	1219.68
1	4	A	.013	.001	.018	-.261	-.021	.011	-.249	-.020	.029	8.11	.75	1218.68
1	5	A	.013	-.001	.018	-.261	.021	.011	-.249	.020	.029	8.11	-.74	1218.70
1	6	A	.011	-.005	.019	-.243	.058	.014	-.232	.053	.033	6.95	-3.28	1219.72
1	7	A	.010	-.004	.184	-.213	.092	-.052	-.203	.088	.132	5.34	-2.11	1196.18
1	8	A	.006	-.001	.013	-.169	.112	.010	-.164	.110	.023	3.58	-.85	1209.72
2	1	A	.009	.004	.020	-.199	-.162	.017	-.190	-.158	.037	5.92	2.74	1222.16
2	2	A	.017	.009	.195	-.262	-.142	-.037	-.246	-.133	.158	8.51	4.51	1211.10
2	3	A	.019	.009	.036	-.315	-.094	.032	-.296	-.085	.068	12.47	5.67	1248.44
2	4	A	.023	.003	.038	-.348	-.035	.031	-.325	-.032	.068	14.65	1.63	1251.64
2	5	A	.023	-.002	.038	-.348	.034	.031	-.325	.032	.069	14.66	-1.60	1251.70
2	6	A	.019	-.009	.036	-.316	.094	.032	-.296	.085	.069	12.49	-5.65	1248.59
2	7	A	.017	-.009	.195	-.263	.142	-.037	-.246	.133	.159	8.53	-4.50	1211.27
2	8	A	.009	-.004	.020	-.199	.162	.017	-.190	.158	.037	5.95	-2.75	1222.29
3	1	A	.011	.008	.034	-.221	-.232	.031	-.210	-.224	.065	7.17	5.30	1244.31
3	2	A	.009	.012	.228	-.298	-.210	.006	-.289	-.198	.234	4.49	5.90	1254.77
3	3	A	-.004	.011	.076	-.341	-.144	.118	-.345	-.133	.194	-2.40	7.36	1315.08
3	4	A	-.005	.005	.087	-.377	-.056	.134	-.382	-.051	.221	-2.93	2.96	1333.65
3	5	A	-.005	-.005	.087	-.377	.055	.134	-.382	.051	.222	-2.98	-2.95	1333.83
3	6	A	-.004	-.012	.076	-.341	.144	.120	-.345	.132	.196	-2.72	-7.39	1315.45
3	7	A	.008	-.011	.228	-.297	.209	.008	-.289	.198	.236	4.12	-5.83	1255.13
3	8	A	.011	-.008	.034	-.221	.232	.032	-.210	.224	.066	7.14	-5.25	1244.66
4	1	A	.010	.004	.055	-.210	-.310	.062	-.200	-.307	.118	6.57	2.34	1280.18
4	2	*	-.082	-.089	.571	-.209	-.244	-.029	-.291	-.333	.542	-15.80	-31.28	1137.40
4	3	*	-.238	-.038	.096	-.050	-.098	.353	-.288	-.136	.449	-152.60	-24.55	1347.82
4	4	*	-.298	.000	.070	-.000	-.075	.380	-.298	-.075	.450	-190.76	.11	1305.32
4	5	*	-.298	-.001	.070	.000	.075	.381	-.298	.074	.451	-190.99	-.39	1305.51
4	6	*	-.243	.035	.093	-.043	.102	.365	-.286	.137	.459	-155.53	22.61	1344.11
4	7	*	-.089	.093	.566	-.202	.238	-.013	-.290	.331	.553	-17.80	33.34	1135.03
4	8	A	.010	-.003	.055	-.209	.309	.066	-.199	.306	.121	6.34	-1.81	1280.74
5	1	A	.013	-.003	.075	-.160	-.374	.083	-.147	-.377	.158	8.03	-1.72	1313.33
5	2	*	-.053	-.352	.498	-.154	-.042	.070	-.207	-.395	.569	-7.22	-112.60	1158.31
5	3	*	-.208	-.288	.013	.017	.119	.263	-.191	-.169	.276	-133.37	-184.85	1209.90
5	4	*	-.282	-.037	-.104	.074	-.061	.255	-.208	-.099	.151	-180.97	-23.84	1014.49
5	5	*	-.282	.036	-.104	.074	.062	.256	-.208	.098	.151	-180.91	23.24	1014.32
5	6	*	-.205	.284	.008	.012	-.113	.278	-.193	.171	.286	-131.22	182.05	1202.01
5	7	*	-.047	.358	.488	-.160	.034	.092	-.208	.392	.580	-5.53	115.73	1153.96
5	8	A	.013	.005	.075	-.160	.371	.089	-.147	.376	.164	8.12	3.14	1313.61
6	1	A	.008	.020	.090	-.065	-.443	.047	-.057	-.423	.137	5.23	12.92	1337.97
6	2	A	.020	-.286	.376	-.100	-.044	-.010	-.081	-.330	.366	10.16	-146.73	1451.79
6	3	A	-.012	-.324	-.103	-.055	.122	.084	-.067	-.202	-.019	-7.83	-207.59	1016.48
6	4	A	-.026	-.045	-.160	-.053	-.064	.059	-.079	-.110	-.101	-16.99	-29.00	921.62
6	5	A	-.026	.045	-.160	-.053	.064	.059	-.079	.109	-.101	-16.83	28.80	921.68
6	6	A	-.011	.324	-.103	-.057	-.122	.086	-.068	.201	-.017	-6.86	207.45	1016.20
6	7	A	.023	.287	.376	-.104	.043	-.004	-.081	.329	.372	11.58	147.01	1452.13
6	8	A	.009	-.020	.090	-.066	.443	.049	-.057	.423	.140	5.46	-12.69	1338.62
7	1	A	-.005	.023	.090	.062	-.450	.031	.057	-.426	.121	-3.50	15.03	1338.27
7	2	A	-.019	-.317	.385	.103	-.007	-.053	.084	-.324	.332	-9.88	-162.74	1463.61
7	3	A	.016	-.352	-.069	.142	.160	.050	.158	-.191	-.019	10.08	-225.35	1073.04
7	4	A	.012	-.047	-.126	.164	-.064	.032	.176	-.111	-.094	7.47	-30.17	979.09
7	5	A	.012	.047	-.126	.164	.063	.032	.176	.110	-.094	7.49	30.17	979.15
7	6	A	.016	.352	-.069	.142	-.161	.051	.158	.191	-.018	10.17	225.39	1073.20
7	7	A	-.019	.317	.385	.104	.007	-.052	.084	.324	.333	-9.85	162.78	1464.02
7	8	A	-.005	-.023	.090	.062	.450	.032	.057	.426	.122	-3.52	-15.04	1338.54

8	1 A	-.013	.019	.074	.170	-.397	.019	.158	-.378	.093	-8.06	12.20	1311.84
8	2 A	.011	-.311	.331	.194	.020	-.059	.205	-.291	.272	5.61	-159.42	1392.03
8	3 A	-.086	-.328	-.035	.106	.156	.022	.020	-.172	-.013	-54.96	-210.33	1130.60
8	4 A	-.021	-.038	-.096	.042	-.062	.017	.021	-.099	-.079	-13.76	-24.08	1028.97
8	5 A	-.021	.038	-.096	.042	.061	.017	.021	.099	-.079	-13.76	24.09	1028.99
8	6 A	-.086	.328	-.035	.106	-.156	.022	.020	.172	-.013	-54.96	210.35	1130.66
8	7 A	.011	.311	.331	.195	-.020	-.058	.206	.291	.273	5.58	159.43	1392.17
8	8 A	-.013	-.019	.074	.170	.397	.020	.158	.378	.094	-8.08	-12.20	1311.94

NO.OF SEAM-- 2

BLOCK COL	BLOCK ROW	CODE	RIDE1	RIDE2	CLOSURE	U1POS	U2POS	U3POS	U1NEG	U2NEG	U3NEG	SIGS1	SIGS2	SIGN
1	1	A	-.000	.002	.009	-.182	-.120	-.022	-.182	-.118	-.013	-.06	1.84	1120.05
1	2	A	-.001	.000	.137	-.231	-.102	-.114	-.232	-.102	.022	-.39	.13	1108.18
1	3	A	.000	-.003	.014	-.272	-.066	-.029	-.272	-.069	-.016	.06	-2.11	1128.90
1	4	A	-.000	-.000	.013	-.296	-.025	-.023	-.296	-.025	-.009	-.12	-.09	1128.35
1	5	A	-.000	.000	.013	-.296	.024	-.023	-.296	.024	-.009	-.12	.09	1128.37
1	6	A	.000	.003	.014	-.272	.066	-.029	-.272	.069	-.016	.07	2.11	1128.94
1	7	A	-.001	-.000	.137	-.232	.102	-.115	-.232	.102	.022	-.38	-.13	1108.23
1	8	A	-.000	-.002	.009	-.183	.120	-.022	-.183	.118	-.013	-.05	-1.85	1120.09
2	1	A	-.000	.003	.015	-.224	-.180	-.034	-.224	-.178	-.019	-.27	2.25	1131.17
2	2	A	.001	.001	.144	-.309	-.167	-.137	-.308	-.166	.007	.60	.37	1120.39
2	3	A	.001	-.003	.025	-.391	-.116	-.053	-.390	-.119	-.028	.70	-2.23	1152.86
2	4	A	.001	-.000	.026	-.443	-.044	-.047	-.441	-.044	-.021	1.06	-.02	1155.36
2	5	A	.001	.000	.026	-.443	.043	-.047	-.441	.043	-.021	1.07	.03	1155.40
2	6	A	.001	.003	.025	-.392	.116	-.053	-.391	.119	-.028	.72	2.23	1152.98
2	7	A	.001	-.001	.144	-.310	.166	-.137	-.309	.166	.007	.62	-.38	1120.52
2	8	A	-.000	-.003	.015	-.224	.181	-.034	-.225	.178	-.019	-.25	-2.26	1131.27
3	1	A	.001	.005	.024	-.269	-.280	-.054	-.268	-.276	-.030	1.03	3.62	1150.34
3	2	A	.008	.002	.169	-.439	-.299	-.179	-.431	-.298	-.010	5.13	1.12	1161.92
3	3	A	.004	-.003	.064	-.645	-.230	-.100	-.640	-.233	-.035	3.58	-2.24	1234.03
3	4	A	.005	.000	.074	-.768	-.085	-.097	-.763	-.085	-.023	3.76	.24	1254.03
3	5	A	.005	-.000	.074	-.768	.084	-.097	-.764	.083	-.023	3.77	-.22	1254.22
3	6	A	.004	.003	.064	-.647	.229	-.100	-.643	.232	-.035	3.58	2.22	1234.74
3	7	A	.008	-.002	.169	-.442	.300	-.179	-.434	.298	-.010	5.12	-1.12	1162.72
3	8	A	.001	-.005	.024	-.270	.281	-.054	-.269	.277	-.030	1.04	-3.64	1150.66
4	1	A	.002	.011	.040	-.287	-.444	-.087	-.285	-.433	-.046	1.48	8.74	1184.48
4	2	*	.007	.012	.718	-.639	-.701	-.493	-.632	-.689	.225	3.18	11.66	1123.09
4	3	*	-.006	-.026	1.591	-1.427	-.506	-.888	-1.433	-.533	.702	-.34	-10.55	1478.90
4	4	*	-.011	-.000	1.804	-1.768	-.178	-.940	-1.780	-.179	.864	-3.11	.14	1541.86
4	5	*	-.011	.000	1.806	-1.770	.175	-.941	-1.781	.175	.865	-3.12	-.16	1542.50
4	6	*	-.007	.026	1.614	-1.439	.496	-.898	-1.446	.522	.716	-.60	10.30	1487.59
4	7	*	.006	-.011	.747	-.653	.711	-.505	-.647	.699	.241	2.94	-11.44	1134.08
4	8	A	.002	-.011	.041	-.289	.447	-.087	-.287	.436	-.046	1.45	-8.68	1185.74
5	1	A	.000	.013	.058	-.218	-.630	-.108	-.218	-.616	-.051	.39	10.80	1220.91
5	2	*	-.005	-.001	1.472	-.511	-1.414	-.857	-.515	-1.415	.614	-.56	7.67	1386.96
5	3	*	-.019	-.078	10.944	-1.250	-1.227	-10.952	-1.269	-1.305	-.008	-.35	-1.44	570.00
5	4	*	-.014	-.003	12.831	-1.631	-.341	-12.754	-1.645	-.344	.077	-.26	-.05	668.27
5	5	*	-.014	.002	12.837	-1.631	.336	-12.759	-1.644	.338	.078	-.25	.04	668.58
5	6	*	-.017	.075	11.018	-1.241	1.210	-11.023	-1.259	1.286	-.005	-.33	1.40	573.86
5	7	*	-.004	.001	1.601	-.498	1.432	-1.076	-.502	1.433	.525	-.35	-7.38	1408.61
5	8	A	.001	-.013	.059	-.218	.635	-.109	-.217	.622	-.050	.42	-10.72	1223.48
6	1	A	.000	.012	.060	-.080	-.725	-.110	-.080	-.713	-.050	.08	9.50	1226.39
6	2	A	.001	-.004	.441	-.181	-1.497	-.334	-.180	-1.501	.107	.49	-2.40	1614.78
6	3	B	.021	-.054	13.549	-.466	-1.449	-13.507	-.444	-1.503	.043	.39	-1.00	705.69
6	4	B	.034	.009	15.721	-.613	-.439	-15.607	-.579	-.430	.114	.63	.17	818.78
6	5	B	.034	-.009	15.724	-.612	.435	-15.610	-.578	.426	.114	.63	-.17	818.95
6	6	B	.022	.053	13.561	-.462	1.447	-13.517	-.440	1.500	.043	.41	.99	706.28
6	7	A	.001	.004	.442	-.175	1.498	-.334	-.174	1.502	.108	.59	2.42	1617.67
6	8	A	.000	-.012	.061	-.078	.727	-.111	-.078	.715	-.050	.08	-9.51	1227.20
7	1	A	-.000	.013	.059	.080	-.737	-.107	.079	-.724	-.048	-.33	10.07	1223.77
7	2	A	-.001	-.000	.453	.125	-1.594	-.340	.124	-1.594	.113	-.37	-.27	1635.10
7	3	B	.028	-.037	14.667	.115	-1.567	-14.635	.142	-1.604	.031	.51	-.69	763.88
7	4	B	.035	.015	17.059	.146	-.466	-16.960	.181	-.451	.098	.66	.29	888.46
7	5	B	.035	-.015	17.060	.147	.464	-16.961	.182	.449	.098	.66	-.28	888.52
7	6	B	.027	.037	14.669	.116	1.566	-14.638	.144	1.604	.031	.51	.69	764.01
7	7	A	-.001	.000	.453	.127	1.594	-.340	.126	1.594	.113	-.41	.27	1635.47
7	8	A	-.000	-.013	.059	.081	.737	-.107	.080	.724	-.048	-.35	-10.09	1223.97

8	1	A	-.004	.010	.049	.235	-.634	-.082	.232	-.624	-.033	-2.83	8.29	1202.97
8	2	A	-.005	.001	.416	.550	-1.388	-.294	.545	-1.387	.121	-3.19	.86	1572.82
8	3	B	.017	-.020	16.381	1.250	-1.357	-16.357	1.267	-1.377	.023	.31	-.36	853.16
8	4	B	.034	.016	18.436	1.589	-.384	-18.362	1.623	-.368	.074	.63	.30	960.20
8	5	B	.034	-.016	18.436	1.589	.383	-18.362	1.623	.367	.074	.63	-.30	960.21
8	6	B	.017	.020	16.381	1.251	1.357	-16.358	1.267	1.377	.023	.31	.37	853.20
8	7	A	-.005	-.001	.416	.551	1.388	-.294	.546	1.387	.121	-3.20	-.86	1572.94
8	8	A	-.004	-.010	.049	.236	.634	-.083	.232	.624	-.033	-2.83	-8.29	1203.05

DISPLACEMENTS AND STRESSES AT OFF-SEAM ELEMENTS.

TOTAL NO. OF OFF-SEAM PLANES -- 1

OFF-SEAM PLANE NO. -- 1

LOCATION OF LOCAL ORIGIN AND GRIDWORK DETAILS --

GRID ORIGIN AT			BLOCK WIDTHS		NO. OF BLOCKS	
XOP	YOP	ZOP	B1	B2	N1	N2
3600.000	4800.000	-13092.000	240.000	240.000	10	1

ORIENTATION OF THE OFF-SEAM PLANE --

EN1X	EN2X	EN3X	EN1Y	EN2Y	EN3Y	EN1Z	EN2Z	EN3Z
1.00000	0.00000	0.00000	0.00000	1.00000	0.00000	0.00000	0.00000	1.00000

N	X	Y	Z	UX	UY	UZ	SXX	SXY	SXZ	SYY	SYZ	SZZ
1	3720.00	4920.00	-13092.00	-.267491	.010594	.390223	587.62	-.73	-81.03	579.60	-.54	1319.30
2	3960.00	4920.00	-13092.00	-.241363	.011075	.436432	565.63	-.60	-105.25	574.21	-.40	1315.69
3	4200.00	4920.00	-13092.00	-.166749	.006675	.415557	522.85	-12.87	-199.07	565.51	7.34	1321.05
4	4440.00	4920.00	-13092.00	-.148187	.009527	.401089	509.03	2.81	-233.31	551.36	.55	1241.55
5	4680.00	4920.00	-13092.00	-.097260	.012183	.340445	489.12	-.09	-313.92	533.05	.87	1141.69
6	4920.00	4920.00	-13092.00	-.071068	.013218	.340386	747.58	-.14	-318.38	707.82	3.69	1226.70
7	5160.00	4920.00	-13092.00	-.073871	.016151	.261491	745.21	3.10	-246.39	690.60	3.61	1117.49
8	5400.00	4920.00	-13092.00	-.061626	.014280	.184492	734.33	-9.52	-205.34	677.55	3.75	1042.34
9	5640.00	4920.00	-13092.00	-.090676	.014213	.128191	699.18	-.56	-126.80	665.31	5.16	1024.97
10	5880.00	4920.00	-13092.00	-.078481	.014904	.094299	668.28	-.58	-108.47	654.53	6.19	1008.72

TIME TAKEN TO OUTPUT THE OFF-SEAM DISPLACEMENTS AND STRESSES -- 2.1140 SECOND

APPENDIX G.--ADAPTATION OF PROGRAM CODE TO OTHER COMPUTERS

The most complex changes to the MULSIM/BM code for implementation on any other computer system would involve the replacement of the CDC mass storage routines (OPENMS, READMS, WRITMS, CLOSMS) with the equivalent IO routines on the user's computer.

Because the CDC Cyber allows up to 10 characters per computer word, any manipulations involving the packing of material property codes and mining status codes would have to be looked at closely and possibly reprogrammed.

The definition of the array MKOD to a character variable type of length 10 would be the easiest way to take care of the 10 characters per word code.

The CDC format editing character for literals is "*". This would have to be changed to the ANSI standard of "'".

The CDC routine to get the elapsed CPU time (second) would have to be replaced with the equivalent routine on the user's computer.

The following routines illustrate how to replace the Cyber mass storage routines:

```
C=================================================================C
C                                                                 C
C                            CLOSMS                               C
C                                                                 C
C=================================================================C
      SUBROUTINE CLOSMS(IUNIT)
C
C     CLOSMS - SIMULATES CDC ROUTINE OF SAME NAME
C
C     ARGUMENTS
C        IUNIT - FILE UNIT NUMBER OF FILE TO BE CLOSED
C
      INTEGER IUNIT
C
      CLOSE(UNIT=IUNIT)
      RETURN
      END
```

```
C=================================================================C
C                                                                 C
C                           OPENMS                                C
C                                                                 C
C=================================================================C
      SUBROUTINE OPENMS(IUNIT,INDEXF,MXREC,IKTYP)
C
C     OPENMS - SIMULATES CDC ROUTINE OF SAME NAME
C              WITH SOME RESTRICTIONS
C     ARGUMENTS
C        IUNIT  - FILE UNIT TO BE OPENED
C        INDEXF - FILE INDEX ARRAY(NOT USED)
C        MXREC  - MAX. NO. OF RECORDS TO BE WRITTEN TO FILE
C                 (NOT USED)
C        IKTYP  - INDEX TYPE FLAG(NOT USED)
C
C     RESTRICTIONS -
C        1.  IF UNIT NOS. ARE THOSE DEFINED FOR RESTART,HISTORY
C            OR GEOMETRY FILES - FILENAME IS SET UP ACCORDINGLY
C        2.  IF UNIT NO. IS NOT DEFINED AS IN 1.  THEN A UNIQUE
C            FILENAME IS GIVEN AS RANF$$%% WHERE $$ AND %% ARE
C            CHOSEN AS FOLLOWS
C            $$ IS A CHARACTER BETWEEN A - Z , AND
C            %% IS THE VALUE IUNIT SUCH THAT A UNIQUE FILENAME
C               IS PRODUCED
C        3.  THE RECORD LENGTH IN BYTES NNREC IS DEFINED
C            ACCORDINGLY
C
      INTEGER IUNIT,INDEXF(1),MXREC,IKTYP
C
      CHARACTER*20 RSTFIL,HSTFIL,GEOFIL
      COMMON/RFILEC/ RSTFIL,HSTFIL,GEOFIL
      INTEGER NRCRST,NRCHST,NRCGEO,NRCFIL,NBYTRL,NBYTIN
      COMMON/RFILEI/ NRCRST,NRCHST,NRCGEO,NRCFIL,NBYTRL,NBYTIN
C
C     LOCAL VARIABLES
C
      CHARACTER*20 FILNAM
      CHARACTER*1 CHAR(26),ICHN(2),ICH
      LOGICAL IEXIST
      INTEGER I,J,INO(2),IDIG,IDIV,NUM,IPOS,NNREC
C
C     INITIALIZE VARIABLES
C
      DATA CHAR/'A','B','C','D','E','F','G','H','I','J','K','L','M'
     1,'N','O','P','Q','R','S','T','U','V','W','X','Y','Z'/
      FILNAM = ' '
C
C     CHECK FOR UNALLOWABLE FILE UNIT
```

```
      IF (IUNIT.LT.0.OR.IUNIT.GT.99) GO TO 10430
C
C     FOR ANY OTHER VALUES OPEN FILE WITH A UNIQUE NAME
C     TRANSFORM TWO DIGIT NO. TO CHARACTERS
C
      IDIG = 1
      IDIV = IUNIT/10
      IF (IDIV.EQ.0) THEN
        INO(1) = IUNIT
      ELSE
        INO(1) = IDIV
        INO(2) = IUNIT-10*IDIV
        IDIG = 2
      ENDIF
C
      DO 1011, J = 1,IDIG
        NUM = INO(J)
        WRITE(ICH,'(I1)')NUM
        ICHN(J) = ICH
 1011 CONTINUE
C
      DO 1032, I = 1,26
        FILNAM(1:4) = 'RANF'
        FILNAM(5:5) = CHAR(I)
C
C       INSERT NUMERIC PART TO FILENAME
C
        IPOS = 5
        DO 1021, J = 1,IDIG
          IPOS = IPOS + 1
          FILNAM(IPOS:IPOS) = ICHN(J)
 1021   CONTINUE
C
C       DOES FILENAME ALREADY EXIST
C
        INQUIRE(FILE=FILNAM,EXIST=IEXIST)
        IF (IEXIST) GOTO 1032
C
        NNREC = NRCFIL
        IF (NNREC.LE.0) GOTO 10430
        GOTO 10510
 1032 CONTINUE
C
10510 CONTINUE
C
C     CONVERT RECORD LENGTH TO BYTES
C
      NNREC = NNREC*NBYTRL
```

```
      OPEN(UNIT=IUNIT,FILE=FILNAM,STATUS='UNKNOWN',ACCESS='DIRECT'
     1,FORM='UNFORMATTED',RECL=NNREC,ERR=10430)
C
10430 CONTINUE
      WRITE(6,80110)IUNIT,NNREC
      STOP
C
80110 FORMAT(1X,'SUBROUTINE OPENMS - CANNOT OPEN FILE'/
     11X,'FILE UNIT NO. =',I5,' REC. LENGTH=',I5/)
      END
C===================================================================C
C                                                                   C
C                             READMS                                C
C                                                                   C
C===================================================================C
      SUBROUTINE READMS(IUNIT,ARR,NOW,JREC)
C
C     READMS - SIMULATES CDC ROUTINE OF SAME NAME
C
C     ARGUMENTS
C        IUNIT - OUTPUT FILE UNIT
C        ARR   - ARRAY TO BE READ
C        NOW   - NUMBER OF ITEMS OF 'A' TO READ
C        JREC  - RECORD NUMBER
C
      INTEGER IUNIT,NOW,JREC
      REAL ARR(NOW)
C
      INTEGER MSTORT
      COMMON/CDCDAT/LARRTP
      INTEGER LARRTP
      PARAMETER (MSTORT=1000)
      COMMON/TMSTOR/XTRRW(MSTORT),NTRRW(MSTORT)
      DOUBLE PRECISION XTRRW
      INTEGER NTRRW
C
C     LOCAL VARIABLES
C
      INTEGER NOV,I
C
C     SET UP NO. OF VARIABLES TO BE TRANSFERRED FOR REALS OR INTEGERS
C     ** ASSUME 2 INTEGERS = 1 REAL **
C     LARRTP .EQ. 0 - FOR AN INTEGER ARRAY
C     LARRTP .NE. 0 - FOR A REAL ARRAY
C
      NOV = NOW
      IF ( LARRTP .EQ. 0) NOV = (NOW+1)/2
      READ (IUNIT,REC=JREC)(ARR(I),I=1,NOV)
C
      RETURN
      END
```

```
C==================================================================C
C                                                                  C
C                          WRITMS                                  C
C                                                                  C
C==================================================================C
      SUBROUTINE WRITMS(IUNIT,ARR,NOW,JREC)
C
C     WRITMS - SIMULATES CDC ROUTINE OF SAME NAME
C
C     ARGUMENTS
C        IUNIT - OUTPUT FILE UNIT
C        ARR   - ARRAY OF VARIABLES TO BE WRITTEN
C        NOW   - NUMBER OF VARIABLES TO BE WRITTEN
C        JREC  - RECORD NUMBER
C
      INTEGER IUNIT,NOW,JREC
      REAL ARR(NOW)
C
      INTEGER MSTORT
      COMMON/CDCDAT/LARRTP
      INTEGER LARRTP
      PARAMETER (MSTORT=1000)
      COMMON/TMSTOR/XTRRW(MSTORT),NTRRW(MSTORT)
      DOUBLE PRECISION XTRRW
      INTEGER NTRRW
C
C     LOCAL VARIABLES
C
      INTEGER NOV,I
C
C     SET UP NO. OF VARIABLES TO BE TRANSFERRED FOR REALS OR INTEGERS
C     ** ASSUME 2 INTEGERS = 1 REAL **
C     LARRTP .EQ. 0 - FOR AN INTEGER ARRAY
C     LARRTP .NE. 0 - FOR A REAL ARRAY
C
      NOV = NOW
      IF (LARRTP .EQ. 0) NOV = (NOW+1)/2
      WRITE(IUNIT,REC=JREC)(ARR(I),I=1,NOW)
C
      RETURN
      END
```

APPENDIX H.--CYBER JOB CONTROL STREAM TO RUN MULSIM/BM

A CDC Cyber procedure file has been written that allows the user to respond to questions about the input files and output files necessary to submit a MULSIM/BM job to the computer.

A sample session using the procedure file follows. The listing of the procedure file itself is included after the sample session.

SAMPLE SESSION

```
READY
GET,PRSUBM/UN=BM0001L
-,PRSUBM

 ENTER PRIORTY JOB PRIORITY(1,3,5) ? 5

  THE JOB PRIORITY ON THE CDC CYBER IS DEFINED AS FOLLOWS:
        1 = DAYTIME PROCESSING AT 10 TIMES
            THE COST OF WEEKEND PROCESSING.
        3   OVERNIGHT PROCESSING AT 2.5 TIMES
            THE COST OF WEEKEND PROCESSING.
        5 = WEEKEND PROCESSING.

 ENTER USRNAM  USER NAME ?  USER1
 ENTER USREXT  USER EXTENSION ? 1234
 ENTER DISCFIL DISPLACEMENTS FILE NAME ? DFILE
 ENTER COFFSFL COEFFICIENTS FILE NAME ? CFILE
 ENTER COFFSDF COEFFICIENTS2 FILE NAME ? CFILE2
 ENTER MLSINPT MULSIM/BM INPUT FILE NAME ? MFILE
 ENTER SEAM1OT SEAM 1 OUTPUT FILE NAME ? SFILE
 ENTER CSEAM1O SEAM 1 COARSE MESH OUTPUT FILE NAME ? CFILE
 ENTER TIMLIMT JOB TIME LIMIT ? 100
 ENTER PRTLIMT JOB PRINT FILE LIMIT ? 1000
 ENTER PRTNAME JOB PRINT OUTPUT FILE NAME ? OFILE
 ENTER DAYNAME JOB DAYFILE FILE NAME ? DFILE
 ENTER JOBDISP JOB OUTPUT DISPOSITION(N,B,E,TO) ? N


  14.33.30.  SUBMIT COMPLETE.  JSN IS AKVZ.
```

After the submit message is displayed, the user may monitor the job execution by entering "ENQ,UJN". The operating system will respond with the status of all jobs submitted by the user.

LISTING OF PROCEDURE FILE PRSUBM

```
.PROC,PRSUBM*I,
PRIORTY" JOB PRIORITY(1,3,5)"=(1,3,5),
USRNAME" USER NAME",
USREXTN" USER EXTENSION",
DISCFIL" DISPLACEMENTS FILE NAME"=(*F),
COFFSFL" COEFFICIENTS FILE NAME"=(*F),
COFFSDF" COEFFICIENTS2 FILE NAME"=(*F),
MLSINPT" MULSIM/BM INPUT FILE NAME"=(*F),
SEAM1OT" SEAM 1 OUTPUT FILE NAME"=(*F),
CSEAM1O" SEAM 1 COARSE MESH OUTPUT FILE NAME"=(*F),
TIMLIMT" JOB TIME LIMIT",
PRTLIMT" JOB PRINT FILE LIMIT",
PRTNAME" JOB PRINT OUTPUT FILE NAME"=(*F),
DAYNAME" JOB DAYFILE FILE NAME"=(*F),
JOBDISP" JOB OUTPUT DISPOSITION(N,B,E,TO)"=(N,B,E,TO).
IF,.NOT.NUM(TIMLIMT),PASS.
REVERT.TIME LIMIT NOT NUMERIC.
ENDIF,PASS.
IF,.NOT.NUM(PRTLIMT),PASS2.
REVERT.PRINT LIMIT NOT NUMERIC.
ENDIF,PASS2.
SUBMIT,FILE,JOBDISP.
.DATA,FILE
/JOB
MULSI,T3600,P=PRIORTY.
/USER
CHARGE,*.
HEADING.$1MINES-DRC
HEADING.$ USRNAME
HEADING.$ USREXTN
GET,BMLBM.
GET,INPUT=MLSINPT.
PURGE,SEAM1OT/NA.
RETURN,SEAM1.
DEFINE,SEAM1=SEAM1OT.
PURGE,CSEAM1O/NA.
RETURN,CSEAM1.
DEFINE,CSEAM1=CSEAM1O.
SETTL(TIMLIMT)
```

```
BMLBM(PL=PRTLIMT).
SKIP,L40.
EXIT.
ENDIF,L40.
REPLACE,DISC=DISCFIL.
SKIP,L41.
EXIT.
PURGE,DISCFIL/NA.
DEFINE,DISCFIL.
REWIND,DISC.
COPYEI,DISC,DISCFIL.
ENDIF,L41.
REPLACE,COFFS=COFFSFL.
SKIP,L42.
EXIT.

PURGE,COFFSFL/NA.
DEFINE,COFFSFL.
REWIND,COFFS.
COPYEI,COFFS,COFFSFL.
ENDIF,L42.
REPLACE,COFFSD=COFFSDF.
SKIP,L50.
EXIT.
PURGE,COFFSDF/NA.
DEFINE,COFFSDF.
REWIND,COFFSD.
COPYEI,COFFSD,COFFSDF.
SKIP,L50.
EXIT.
ENDIF,L50.
PURGE,PRTNAME/NA.
DEFINE,PRTNAME.
REWIND,OUTPUT.
SKIPR,OUTPUT,4.
COPYEI,OUTPUT,PRTNAME.
RETURN,PRTNAME.
SKIP,L60.
EXIT.
ENDIF,L60.
DAYFILE,DAYNAME.
REPLACE,DAYNAME.
/EOF
REVERT,NOLIST.
```

APPENDIX I.--ENGINEERING SYMBOLS USED IN THIS REPORT

A	element area, in^2
C	element closure, in
c	initial gob-zone closure, in
E_g	secant elastic gob modulus (linear modulus), psi
E_M	modulus of deformation, GPa
F	normal force acting on an element, lb
g	gob-zone height, in
h	element height, in
k'	gob material parameter, psi
M	linear elastic gob modulus (input to program), psi
m	extracted seam height, in
P	equilibrium point
p	gob resistance, psi
S	area stiffness (stiffness per unit area), lb/in^3
S_g	area stiffness of gob, lb/in^3
t	seam thickness (input to program), in
ε	strain, in/in
ε_g	strain sustained by gob, in/in
ε_{max}	maximum possible gob strain, in/in
ε_o	equilibrium gob strain (at virgin overburden normal stress), in/in

σ	stress, psi
σ_g	stress sustained by gob (linear equation), psi
σ_{NL}	stress sustained by gob (nonlinear equation), psi
σ_o	virgin overburden normal stress, psi
σ_{ij}	stress acting in the positive X_j direction on a plane whose outward normal points in the negative X_i direction
ω	gob zone convergence, main roof to floor, in
μ_i	displacement in the positive X_i direction
X_i	global axis in the i-direction
$X_{\bar{i}}$	local axis, or rotated axis, in the i-direction
X	X_1 or $X_{\bar{1}}$, as appropriate
Y	X_2 or $X_{\bar{2}}$, as appropriate
Z	X_3 (depth below surface; numerically negative)

U.S. GOVERNMENT PRINTING OFFICE: 1988 — 505-016/80,007

INT.-BU.OF MINES,PGH.,PA. 28612